OLITEHNICA” TIMIŞOARA
ECANICĂ
ă Tehnică şi Vibraţii

Ing. Cornel POPA

TEZĂ DE DOCTORAT

- Anexe -

Conducător ştiinţific:
Prof.univ.Dr.Ing. Liviu BRÎNDEU

- 2002 -

STUDIU PRIVIND INFLUENȚA VIBRAȚIILOR ASUPRA COMPORTĂRII MECANICE A ORGANISMULUI UMAN

ANEXA 3.1

Codul sursă al programului software *antet.pas* este redat în cele ce urmează. Acest program este scris în limbajul de programare Pascal.

```
program antetul;
uses crt;
var ch:char;
procedure pres_a_key;
   begin
      writeln;
      gotoxy(25,24);
      writeln(' >>> APASATI ORICE TASTA PENTRU A CONTINUA <<< ');
      writeln;
      ch:=readkey;
   end;
procedure Introducere;
 const A:array[1..9] of
integer=(500,800,200,800,500,200,500,800,500);
 var i,j:integer;
      s:string;
      c:char;
  begin
   TextBackGround(7);
   ClrScr;
   TextColor(Yellow);
   for i:=1 to 10 do
    begin
     GotoXY(25,3+i);
     write('ŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰ');
    end;
   window(10,5,72,12);
   TextColor(Yellow);
   TextBackGround(0);
   ClrScr;
   writeln;
   writeln;
   writeln;
   writeln('       S I S T E M    E X P E R T   -   S I M P T O v i
b');
   window(1,1,80,25);
   TextColor(Yellow);
   TextBackGround(7);
   GotoXY(3,20);
   writeln('  Profesor coordonator: Prof.univ.dr.ing.Liviu
BRINDEU');
   GotoXY(16,22);
   writeln('                               Doctorand: ing. Cornel
POPA');
   readln;
   ClrScr;
   window(32,5,47,7);
   TextColor(Yellow);
   TextBackGround(Red);
   ClrScr;
   writeln;
   s:='H E L L O  !';
   j:=1;
   while j<>13 do
   begin
   for i:=13 downto j do
    begin
     GotoXY(3+i,2);
     write(s[j]);
     if s[j]<>' 'then delay(20);
     GotoXY(3+i,2);
     write(' ');
    end;
    GotoXY(3+i,2);
    write(s[j]);
    if s[j]<>' ' then
     begin
          sound(2096+2*j);
          delay(5);
          nosound;
      end;
    j:=j+1;
   end;
   delay(400);
   window(10,5,70,18);
   TextColor(14);
   TextBackGround(Blue);
   ClrScr;
   writeln;
   writeln;
   writeln('    Tema abordata face parte din domeniul Inteligentei
ar-');
   writeln(' tificiale.');
   writeln('    Programul incearca sa faca "diagnosticarea"
frecventei');
   writeln(' vibratiilor functie de simptomele provocate de catre
aces-');
   writeln(' tea asupra subiectilor umani. ');
   writeln('    Astfel se incearca sa se specifice domeniul de
frecventa');
   writeln(' in care au loc vibratii, functie de raspunsurile pe
care le');
   writeln(' primeste de la operator (utilizatorul acestui program),
a-');
   writeln(' nalizand baza de cunostinte inregistrata in fisierul
numit');
```

```
   writeln(' simptome.dat');
   window(1,1,80,25);
   TextColor(14);
   TextBackGround(1);
   pres_a_key;
   clrscr;
end;

begin
  introducere;
  gotoxy(10,10);
end.
```

ANEXA 3.2

Codul sursă al programului de calculator *simptome.cpp* este redat în continuare. Acest program este scris în limbajul de programare BorlandC.

```
#include <stdio.h>
#include <alloc.h>
#include <conio.h>
#include <ctype.h>
#include <string.h>
#include <dos.h>

#define MAX 100

typedef struct attribute{
                 char attrib[80];
                 struct attribute *next;
} at;

typedef struct object{
                 char name[80];
                 struct attribute *alist;
} ob;

typedef struct rejected_object{
           char name[80];
           char attrib[80];
           char condition;
} rj;

rj r_base[100];

struct object k_base[MAX];
int l_pos=-1;
int r_pos=-1;
struct attribute *yes,*no;
struct attribute *yesnext,*nonext;
void free_trails(void);
void enter(void);
void query(void);
char menu(void);
void save(void);
void load(void);
int get_next(void);
short tryy(struct attribute *p,char *ob);
int trailno(struct attribute *p,char *ob);
int trailyes(struct attribute *p, char*ob);
int ask(char *attrib);
void reject(char *ob,char *at,char cond);
void reasoning(char *ob);
```

```
int is_in(char c,char *s);
void clear_kbase(void);
void tratament(void);
void final(void);
void initial(void);
void cuprins(void);
void frecventa(void);
void simpto1(void);
void simpto2(void);
void simpto3(void);
void simpto4(void);
void simpto5(void);
void simpto6(void);
void simpto7(void);
void simpto8(void);
void simpto9(void);
void simpto0(void);

void main(void)
{
       char ch,ch1,frecventa;
       clrscr();
       window(1,1,80,80);
       textbackground(1);
       textcolor(14);
       cprintf("\n");
       no=yes='\0';
       do {
                    free_trails();
                    ch=menu();
                    switch(ch)
                    {
                         case 'e':enter(); break;
                         case 'q':query(); break;
                         case 's':save(); break;
                         case 'l':load();break;
                         case 'c':cuprins();
                    }
            }while (ch!='x');
       final();
}

void free_trails()
{
          struct attribute *p;
          while(yes)
          {
                    p=yes->next;
                    free(yes);
                    yes=p;
          }
          while(no)
          {
                    p=no->next;
                    free(no);
                    no=p;
          }
          r_pos=-1;
}

void enter()
{
          int t,i;
          struct attribute *p,*oldp;
          for(;;)
          {
                t=get_next();
                if(t==-1)
                {
                          printf("Out if list space.\n");
                          return;
                }
                printf("\n\n\nDomeniul de frecvente:  ");
                gets(k_base[t].name);
                if(!*k_base[t].name){
                              l_pos--;
                              break;
                }
                p=(struct attribute *) malloc(sizeof(at));
                if (p=='\0')
                {
                           printf("Out of memory");
                           return;
                }
                k_base[t].alist=p;
                for(i=0;i<sizeof(p->attrib);i++)
                               p->attrib[i]=' ';
                printf("\nSimptome .......(Apasa <ENTER> pt.end)
\n");
                for(;;)
                {
                          printf("       : ");
                          gets(p->attrib);
                          if(!p->attrib[0])
                                   break;
                          oldp=p;
                          p->next=(struct attribute *)
malloc(sizeof(at));
                          if(p->next=='\0')
                          {
                                   printf("Out of memory .\n");
                                   return;
                          }
                          p=p->next;
                          p->next='\0';
                          for(i=0;i<sizeof(p->attrib);i++)
                               p->attrib[i]=' ';
                 }
                 oldp->next='\0';
          }
}

void query()
{
          int t,i;
          char ch,chh;
          struct attribute *p;
          for(t=0;t<=l_pos;t++)
          {
               p=k_base[t].alist;
               if(tryy(p,k_base[t].name))
               {
                 textcolor(13);
                 cprintf("\n\
ŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰ
ŰŰŰŰŰŰŰŰŰ");
                 printf("\n\n  < %s > se potriveste descrierii
facute \n\n",k_base[t].name);
                 cprintf("
ŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰŰ
ŰŰŰŰŰŰŰŰŰ");
                 textcolor(14);
                 cprintf("\n\n   << Doresti sa afli informatii
suplimentare despre");
                 printf("\n       %s  ? >>  (d/n)",k_base[t].name);
                 chh=tolower(getche());
                 if (chh=='d')
                 {
                          if (!strcmp(k_base[t].name,"Domeniul 4-6
Hz"))
                                        simpto1();
                          if (!strcmp(k_base[t].name,"Domeniul 6-7
Hz"))
                                        simpto2();
                          if (!strcmp(k_base[t].name,"Domeniul 7-8
Hz"))
                                        simpto3();
                          if (!strcmp(k_base[t].name,"Domeniul 8-9
Hz"))
                                        simpto4();
                          if (!strcmp(k_base[t].name,"Domeniul 9-10
Hz"))
                                        simpto5();
                          if (!strcmp(k_base[t].name,"Domeniul 10-12
Hz"))
                                        simpto6();
                          if (!strcmp(k_base[t].name,"Domeniul 12-13
Hz"))
                                        simpto7();
                          if (!strcmp(k_base[t].name,"Domeniul 13-16
Hz"))
                                        simpto8();
                          if (!strcmp(k_base[t].name,"Domeniul 16-18
Hz"))
                                        simpto9();
                          if (!strcmp(k_base[t].name,"Domeniul 18-20
Hz"))
                                        simpto0();

                    }
//   cprintf("\n\n\n     Continuati ?    (y/n) : ");
//   ch=tolower(getche());
//   printf("\n");
//   if (ch=='n')
                    return;
                }
           }
           delay(1000);
           textbackground(0);
           textcolor(140);
           cprintf("\n\n\n\n");
           printf("\n
ÉÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍ»");
           printf("\n     ş Consulta si un expert uman pentru
lamuriri/alte lamuriri !ş");
           printf("\n
ČÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍĽ\n");
           cprlntf("\n\n\n");
           textbackground(1);
           textcolor(14);
           delay(3000);
           clrscr();
}

short tryy(attribute *p, char *ob)
{
          char answer;
          struct attribute *a;
          if(!trailno(p,ob))      return 0;
          if(!trailyes(p,ob))      return 0;
          while(p)
          {
                    if(ask(p->attrib))
                    {
```

```
                               printf("\n\n  Atribut(y/n/w):  %s ?
",p->attrib);

                               answer=tolower(getche());
                               printf("\n");
                               a=(struct attribute *)

malloc(sizeof(at));
                               if(!a)
                               {
                                     printf("Out of memory\n");
                                     return 0;
                               }
                               a->next='\0';
                               switch(answer)
                               {
                           case 'n':  strcpy(a->attrib,p->attrib);
                                        if(!no)
                                         {
                                             no=a;
                                             nonext=no;
                                         }
                                         else
                                         {
                                           nonext->next=a;
                                           nonext=a;
                                         }
                                           reject(ob,p->attrib,'n');
                                           return 0;
                           case 'y':  strcpy(a->attrib,p->attrib);
                                         if(!yes)
                                         {
                                             yes=a;
                                             yesnext=yes;
                                         }
                                         else
                                         {
                                             yesnext->next=a;
                                             yesnext=a;
                                         }
                                         p=p->next;
                                         break;
                           case 'w':
                                         reasoning(ob);
                                         break;
                               }
                    }
                    else
                           p=p->next;
          }
          return 1;
}
```

```
int trailno(struct attribute *p,char *ob)
{
          struct attribute *a,*t;
          a=no;
          while(a)
          {
                    t=p;
                    while(t)
                    {
                          if(!strcmp(t->attrib,a->attrib))
                          {
                                   reject(ob,t->attrib,'n');
                                   return 0;
                          }
                          t=t->next;
                    }
                    a=a->next;
          }
          return 1;
}

int trailyes(struct attribute *p,char *ob){
       struct attribute *a,*t;
       char ok;
       a=yes;
       while(a)
       {
                ok=0;
                t=p;
                while(t)
                {
                         if(!strcmp(t->attrib,a->attrib))
                         {
                               ok=1;
                         }
                         t=t->next;
                }
                if(!ok)
                {
                      reject(ob,a->attrib,'y');
                      return 0;
                }
                a=a->next;
      }
      return 1;
}

int ask(char *attrib)
{
       struct attribute *p;
       p=yes;
       while(p && strcmp(attrib,p->attrib))
                 p=p->next;
       if(!p)
                 return 1;
       else
                 return 0;
}

void reasoning(char *ob)
{
      struct attribute *t;
      int i;
      textbackground(2);
      textcolor(10);
      cprintf("\n
ßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßß
ßßßßßßßßßß");
      printf("\n");
      cprintf("  Posibil ............ %s\n\n",ob);
      if(yes)
            printf("\n   Simptome predominante:\n");
      t=yes;
      while(t)
      {
            printf("          %s\n",t->attrib);
            t=t->next;
      }
      for(i=0;i<=r_pos;i++)
      {
            printf("\n ' %s ' nu se potriveste deoarece
",r_base[i].name);
            if(r_base[i].condition=='n')
                     printf(" ' %s ' nu este un simptom al acestei
frecvente de vibratie!\n ",r_base[i].attrib);
            else
                     printf("\n%s  este un simptom potrivit   . \n
",r_base[i].attrib);
       }
       printf("
ÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜ
ÜÜÜÜÜÜÜÜÜÜ");
       textbackground(1);
       textcolor(14);
       cprintf("\n");
}

void reject(char *ob,char *at,char cond)
{
      r_pos++;
      strcpy(r_base[r_pos].name,ob);
      strcpy(r_base[r_pos].attrib,at);
      r_base[r_pos].condition=cond;
}

int get_next()
{
           l_pos++;
           if(l_pos<MAX)
                      return l_pos;
           else
                      return -1;
}

char menu(void)
{
          char ch;
          printf("\n\n\n\n\n           M E N I U\n");
          printf("               ---------\n\n");
          printf("   * E *    ......Introducere date\n");
          printf("   * Q *    ......Diagnoza\n");
          printf("   * S *    ......Salveaza datele inscrise prin 
\n");
          printf("                               optiunea 
<Introducere date>\n");
          printf("   * L *    ......Incarca baza de cunostinte\n");
          printf("   * C *    ......Cuprins si alte detalii\n");
          printf("   * X *    ......Iesire\n\n");
          do{
                    printf("\n\n\n\n ALEGE UNUL DIN CAZURILE DE MAI 
SUS:");
                    ch=tolower(getche());
               }while  (!is_in(ch,"eqslcx"));
          printf("\n\n\n\n");
          return ch;
}

void save()
{
       int t,x;
       struct attribute *p;
       FILE *fp;
       if((fp=fopen("simptom1.dat","a"))==0)   //pentru completarea 
BD
       {                                        //se scrie 
"simptome.dat","a"
                printf("  Fisierul nu poate fi deschis\n");
                return;
       }
       printf(" Salvarea bazei de cunostinte \n");
       for(t=0;t<=1;++t)
       {
                for(x=0;x<sizeof(k_base[t].name);x++)
```

```
                      putc(k_base[t].name[x],fp);
                p=k_base[t].alist;
                while(p)
                {
                         for(x=0;x<sizeof(p->attrib);x++)
                                   putc(p->attrib[x],fp);
                         p=p->next;
                }
                for(x=0;x<sizeof(p->attrib);x++)
                         putc('\0',fp);
          }
          putc(0,fp);
          fclose(fp);
}

void load()
{
       int t,x;
       struct attribute *p,*oldp;
       FILE *fp;
       if((fp=fopen("simptome.dat","r"))==0)
       {
                printf(" Fisierul nu poate fi deschis !\n");
                return;
       }
       printf(" Incarcarea bazei de cunostinte \n");
       clear_kbase();
       for(t=0;t<MAX;t++)
       {
                if((k_base[t].name[0]=getc(fp))==0)
                           break;
                for(x=1;x<sizeof(k_base[t].name);x++)
                          k_base[t].name[x]=getc(fp);
                k_base[t].alist=(struct attribute *)
malloc(sizeof(at));
                p=k_base[t].alist;
                if(!p)
                {
                          printf("Out of memory \n");
                          return;
                }
                memset(p,0,sizeof(at));
                oldp=p;
                for(;;)
                {
                                    for(x=0;x<sizeof(p->attrib);x++)
                                              p->attrib[x]=getc(fp);

                                    if(!p->attrib[0])
                                    {
                                             if(oldp->next)
```

```
                                              {
                                                   free(oldp->next);
                                                   oldp->next=NULL;
                                              }
                                              break;
                                     }
                                     p->next=(struct attribute *)
malloc(sizeof(at));
                                     if(!p->next)
                                     {
                                         printf("Out of memory\n");
                                         return;
                                     }
                                     oldp=p;
                                     p=p->next;
                }
       }
       fclose(fp);
       l_pos=t-1;
}

void clear_kbase()
{
          int t;
          struct attribute *p,*p2;
          for(t=0;t<l_pos;t++)
          {
                p=k_base[t].alist;
                while(p)
                {
                           p2=p->next;
                           free(p);
                           p=p2;
                }
          }
}

int is_in(char ch,char *s)
{
       while(*s)
               if(ch==*s++)
                      return 1;
       return 0;
}

void cuprins()
{
          char boala;
          printf("\n\n      DOMENIUL DE FRECVENTE\n");
          printf("      ---------------------\n\n");
          printf("   * 1 *   Frecventa  4- 6 Hz\n");
          printf("   * 2 *   Frecventa  6- 7 Hz\n");
          printf("   * 3 *   Frecventa  7- 8 Hz\n");
          printf("   * 4 *   Frecventa  8- 9 Hz\n");
          printf("   * 5 *   Frecventa  9-10 Hz\n");
          printf("   * 6 *   Frecventa 10-12 Hz\n");
          printf("   * 7 *   Frecventa 12-13 Hz\n");
          printf("   * 8 *   Frecventa 13-16 Hz\n");
          printf("   * 9 *   Frecventa 16-18 Hz\n");
          printf("   * 0 *   Frecventa 18-20 Hz\n");
          printf("   * X *   Exit ... Iesire\n");
          do{
                    printf("\nALEGE UNUL DIN CAZURILE DE MAI SUS:
");
                    boala=tolower(getche());
               }while  (!is_in(boala,"1234567890x"));
          printf("\n");
          switch(boala)
          {
                 case '1' : simpto1();break;
                 case '2' : simpto2();break;
                 case '3' : simpto3();break;
                 case '4' : simpto4();break;
                 case '5' : simpto5();break;
                 case '6' : simpto6();break;
                 case '7' : simpto7();break;
                 case '8' : simpto8();break;
                 case '9' : simpto9();break;
                 case '0' : simpto0();break;
          }
}

void final()
{
     clrscr();
     gotoxy(5,12);
     textbackground(3);
     textcolor(11);
     cprintf("VA  MULTUMIM  PENTRU  CA  ATI  CONSULTAT \n");
     cprintf(" ACEST   SISTEM   EXPERT !\n\n");
     gotoxy(45,18);
     printf("inginer profesor: Cornel Popa");
     delay(2000);
}

void simpto1(void)
{
  char ch1;
  textbackground(6);
  textcolor(14);
  cprintf("\n\n");
  printf("\n               Frecvente de 4-6 Hz  simptome\n");
  printf("               ==============================\n\n");
  textbackground(1);
  printf("\n      - sentiment general de stinghereala");
  printf("\n      - contractia muschilor");
  printf("\n      - influente asupra miscarii respiratorii");
  printf("\n      - dureri ale abdomenului");
  printf("\n      - dureri ale cosului pieptului");
  printf("\n\n\n\n");
  printf("\n                < DORESTI SA AFLI INFORMATII
SUPLIMENTARE ? D/N  >");
  ch1=tolower(getche());
                    printf("\n");
                    if (ch1=='n') return;
  printf("\n\n     \n");
  printf(" ............................... \n");
  printf("\n- trimite un e-mail la adresa:\n");
  printf("\n ...............................\n ");
  printf("\n\n           cpopa@aut.utt.ro\n");
  printf("\n ...............................\n ");
  cprintf("\n");
  ch1=tolower(getche());
}

void simpto2(void)
{
  char ch1;
  textbackground(6);
  textcolor(14);
  cprintf("\n\n");
  printf("\n                Frecvente de 6-7 Hz  simptome\n");
  printf("               ==============================\n\n");
  textbackground(1);
  printf("\n      - sentiment general de stinghereala");
  printf("\n      - contractia muschilor");
  printf("\n      - influente asupra miscarii respiratorii");
  printf("\n      - dureri ale abdomenului");
  printf("\n      - dureri ale cosului pieptului");
  printf("\n      - simptome la maxilarul inferior");
  printf("\n\n");
  printf("\n        < DORESTI SA AFLI INFORMATII SUPLIMENTARE ? D/N
>");
  ch1=tolower(getche());
                    printf("\n");
                    if (ch1=='n') return;
  printf("\n\n     \n");
  printf(" ............................... \n\n");
  printf("\n- trimite un e-mail la adresa:\n");
  printf("\n ...............................\n ");
  printf("\n\n           cpopa@aut.utt.ro\n");
  printf("\n ...............................\n ");
  cprintf("\n");
  ch1=tolower(getche());
}

void simpto3(void)
{
  char ch1;
  textbackground(6);
  textcolor(14);
  cprintf("\n\n");
  printf("\n               Frecvente de 7-8 Hz  simptome\n");
  printf("               =============================\n\n");
  textbackground(1);
  printf("\n      - sentiment general de stinghereala");
  printf("\n      - contractia muschilor");
  printf("\n      - influente asupra miscarii respiratorii");
  printf("\n      - dureri ale abdomenului");
  printf("\n\n\n\n");
  printf("\n        <  DORESTI SA AFLI INFORMATII SUPLIMENTARE ? D/N
>");
  ch1=tolower(getche());
                      printf("\n");
                      if (ch1=='n') return;
  printf("\n\n     \n");
  printf(" ............................... \n\n");
  printf("\n- trimite un e-mail la adresa:\n");
  printf("\n ...............................\n ");
  printf("\n\n           cpopa@aut.utt.ro\n");
  printf("\n ...............................\n ");
  cprintf("\n");
  ch1=tolower(getche());
}

void simpto4(void)
{
  char ch1;
  textbackground(6);
  textcolor(14);
  cprintf("\n\n");
  printf("\n               Frecvente de 8-9 Hz  simptome\n");
  printf("               =============================\n\n");
  textbackground(1);
  printf("\n      - sentiment general de stinghereala");
  printf("\n      - contractia muschilor");
  printf("\n      - dureri ale abdomenului");
  printf("\n\n\n\n");
  printf("\n        <  DORESTI SA AFLI INFORMATII SUPLIMENTARE ? D/N
>");
  ch1=tolower(getche());
                      printf("\n");
                      if (ch1=='n') return;
  printf("\n\n     \n");
  printf(" ............................... \n\n");
  printf("\n- trimite un e-mail la adresa:\n");
  printf("\n ...............................\n ");
  printf("\n\n           cpopa@aut.utt.ro\n");
  printf("\n ...............................\n ");
  cprintf("\n");
  ch1=tolower(getche());
}

void simpto5(void)
{
  char ch1;
  textbackground(6);
  textcolor(14);
  cprintf("\n\n");
  printf("\n               Frecvente de 9-10 Hz  simptome\n");
  printf("               ==============================\n\n");
  textbackground(1);
  printf("\n      - dureri ale abdomenului");
  printf("\n      - dorinta de a urina");
  printf("\n\n\n\n");
  printf("\n        <  DORESTI SA AFLI INFORMATII SUPLIMENTARE ? D/N
>");
  ch1=tolower(getche());
                      printf("\n");
                      if (ch1=='n') return;
  printf("\n\n     \n");
  printf(" ............................... \n\n");
  printf("\n- trimite un e-mail la adresa:\n");
  printf("\n ...............................\n ");
  printf("\n\n           cpopa@aut.utt.ro\n");
  printf("\n ...............................\n ");
  cprintf("\n");
  ch1=tolower(getche());
}

void simpto6(void)
{
  char ch1;
  textbackground(6);
  textcolor(14);
  cprintf("\n\n");
  printf("\n               Frecvente de 10-12 Hz  simptome\n");
  printf("               ===============================\n\n");
  textbackground(1);
  printf("\n      - dorinta de a urina");
  printf("\n\n\n\n");
  printf("\n        <  DORESTI SA AFLI INFORMATII SUPLIMENTARE ? D/N
>");
  ch1=tolower(getche());
                      printf("\n");
                      if (ch1=='n') return;
  printf("\n\n     \n");
  printf(" ............................... \n\n");
  printf("\n- trimite un e-mail la adresa:\n");
  printf("\n ...............................\n ");
  printf("\n\n           cpopa@aut.utt.ro\n");
  printf("\n ...............................\n ");
  cprintf("\n");
  ch1=tolower(getche());
}

void simpto7(void)
{
  char ch1;
  textbackground(6);
  textcolor(14);
  cprintf("\n\n");
  printf("\n               Frecvente de 12-13 Hz  simptome\n");
  printf("               ===============================\n\n");
  textbackground(1);
  printf("\n      - nod in gat");
  printf("\n      - dorinta de a urina");
  printf("\n\n\n\n");
  printf("\n        <  DORESTI SA AFLI INFORMATII SUPLIMENTARE ? D/N
>");
  ch1=tolower(getche());
                      printf("\n");
                      if (ch1=='n') return;
  printf("\n\n     \n");
  printf(" ............................... \n\n");
  printf("\n- trimite un e-mail la adresa:\n");
  printf("\n ...............................\n ");
  printf("\n\n           cpopa@aut.utt.ro\n");
  printf("\n ...............................\n ");
  cprintf("\n");
  ch1=tolower(getche());
}

void simpto8(void)
{
  char ch1;
  textbackground(6);
  textcolor(14);
  cprintf("\n\n");
  printf("\n               Frecvente de 13-16 Hz  simptome\n");
  printf("               ===============================\n\n");
  textbackground(1);
  printf("\n      - simptome la cap");
  printf("\n      - influente asupra vorbirii");
  printf("\n      - nod in gat");
  printf("\n      - dorinta de a urina");
  printf("\n      - tonus muscular nedorit");
  printf("\n\n\n\n");
  printf("\n        <  DORESTI SA AFLI INFORMATII SUPLIMENTARE ? D/N
>");
  ch1=tolower(getche());
                      printf("\n");
                      if (ch1=='n') return;
  printf("\n\n     \n");
  printf(" ............................... \n\n");
  printf("\n- trimite un e-mail la adresa:\n");
  printf("\n ...............................\n ");
  printf("\n\n           cpopa@aut.utt.ro\n");
  printf("\n ...............................\n ");
  cprintf("\n");
  ch1=tolower(getche());
}

void simpto9(void)
{
  char ch1;
  textbackground(6);
  textcolor(14);
  cprintf("\n\n");
  printf("\n               Frecvente de 13-16 Hz  simptome\n");
  printf("               ================================\n\n");
  textbackground(1);
  printf("\n      - simptome la cap");
  printf("\n      - influente asupra vorbirii");
  printf("\n      - dorinta de a urina");
  printf("\n      - tonus muscular nedorit");
  printf("\n\n\n\n");
  printf("\n        <  DORESTI SA AFLI INFORMATII SUPLIMENTARE ? D/N
>");
  ch1=tolower(getche());
                      printf("\n");
                      if (ch1=='n') return;
  printf("\n\n     \n");
  printf(" ............................... \n\n");
  printf("\n- trimite un e-mail la adresa:\n");
  printf("\n ...............................\n ");
  printf("\n\n           cpopa@aut.utt.ro\n");
  printf("\n ...............................\n ");
  cprintf("\n");
  ch1=tolower(getche());
}

void simpto0(void)
{
  char ch1;
  textbackground(6);
  textcolor(14);
  cprintf("\n\n");
  printf("\n               Frecvente de 18-20 Hz  simptome\n");
  printf("               ================================\n\n");
  textbackground(1);
  printf("\n      - simptome la cap");
  printf("\n      - influente asupra vorbirii");
  printf("\n      - tonus muscular nedorit");
  printf("\n\n\n\n\n");
  printf("\n        <  DORESTI SA AFLI INFORMATII SUPLIMENTARE ? D/N
>");
  ch1=tolower(getche());
                      printf("\n");
                      if (ch1=='n') return;
  printf("\n\n     \n");
  printf(" .............................. \n\n");
  printf("\n- trimite un e-mail la adresa:\n");
  printf("\n ...............................\n ");
  printf("\n\n           cpopa@aut.utt.ro\n");
  printf("\n ...............................\n ");
  cprintf("\n");
  ch1=tolower(getche());
}
```

ANEXA 4.1

Codul sursă al acestui programului *elastfin.pas* este redat în continuare:

```
program estimare_constante_elastice;
uses crt,printer;
var men1,men2:integer;
    m,s1,m1,m2,m3,m4,m5,m6,m7,r:real;
    pr1,pr2,pr3,pr4,pr5,pr6,pr7,pr:real;
    fn1,fn2,fn3,fn4,fn5,fn6,fn7,a,b,c:real;
  omega1,omega2,omega3,omega4,omega5,omega6,omega7:real;
    k12,k23,k34,k35,k45,k56,k57,k1f:real;
    ch,vm,vp,vf,vimp,cp,st,im:char;

procedure tasta1;
     begin
          ch:=readkey;
     end;

procedure antet;
     begin
          textbackground(3);
          clrscr;
          textbackground(1);
          textcolor(3);
          gotoxy(20,10);
          writeln('  M  O  D  E  L    M  E  C  A  N  I  C');
          gotoxy(10,12);
          writeln('al corpului omenesc asezat Œn picioare pe o
platform†');
          writeln;
          delay(2050);
          clrscr;
          textbackground(3);
          textcolor(1);
          gotoxy(12,3);
          writeln(' D E T E R M I N A R E  C O N S T A N T E  E
L A S T I C E');
          writeln;
          writeln;
          tasta1;
     end;

procedure tasta;
     begin
          gotoxy(23,20);
          writeln('>> APASATI ORICE TASTA PENTRU A COMTINUA! <<');
          ch:=readkey;
     end;

procedure meniu1;
     begin
          writeln('       D A T E   I N I T I A L E');
          writeln('       ~~~~~~~~~~~~~~~~~~~~~~~~~');
          writeln('  Tasteaz†:');
          writeln;
          writeln('     1 ..... pentru folosirea datelor
implicite');
          writeln;
          writeln('     2 ..... pentru a introduce datele de la
tastatur†');
          writeln;
          readln(men1);
          writeln;
          writeln;
     end;

procedure meniu2;
     begin
          writeln('    Tasteaz†:');
          writeln;
          writeln('      1 ..... Dati masa total† a corpului');
          writeln;
          writeln('      2 ..... Dati masele celor 7 elemente
componente');
          writeln;
          readln(men2);
          writeln;
     end;

procedure masa1;
     begin
          clrscr;
          writeln('Dati masa total† a corpului, exprimat† Œn kg
!');
          write('        m=');
          readln(m);
          writeln;
          m1:=7.3/100*m;
          m2:=15/100*m;
          m3:=19/100*m;
          m4:=9.7/100*m;
          m5:=36.5/100*m;
          m6:=7/100*m;
          m7:=5.5/100*m;
          write('Doresti s† vezi valorile maselor componente?
(D/N)');
          readln(vm);
```

```
          writeln;
          if (vm='d') or (vm='D') then
             begin
                  writeln;
                  writeln('Masele elementelor modelului sunt:');
                  writeln;
                  writeln('     m1=',m1:9:3,' kg');
                  writeln('     m2=',m2:9:3,' kg');
                  writeln('     m3=',m3:9:3,' kg');
                  writeln('     m4=',m4:9:3,' kg');
                  writeln('     m5=',m5:9:3,' kg');
                  writeln('     m6=',m6:9:3,' kg');
                  writeln('     m7=',m7:9:3,' kg');
                  tasta;
             end;
     end;

procedure masa2;
     begin
          clrscr;
          writeln('Dati masele elementelor componente ale corpului,
Śn kg !');
          writeln;
          write('      m1=');readln(m1);
          write('      m2=');readln(m2);
          write('      m3=');readln(m3);
          write('      m4=');readln(m4);
          write('      m5=');readln(m5);
          write('      m6=');readln(m6);
          write('      m7=');readln(m7);
          writeln;
          m:=m1+m2+m3+m4+m5+m6+m7;
          delay(850);
          writeln('Masa total† a corpului omenesc este: m=',m:9:3,'
kg');
          writeln;
          tasta1;
     end;

procedure frecventa;
     begin
          writeln;
          writeln('Doresti s† dai valorile frecventelor de rezonant†
ale');
          write('    maselor componente ale modelului mecanic ?
(D/N)');
          readln(vf);
          writeln;
          if (vf='d') or (vf='D') then
             begin
                 writeln('Dati frecventele de rezonant†, Śn Hz!');
                 writeln;
                 write('             fn1:=');readln(fn1);
                 write('             fn2:=');readln(fn2);
                 write('             fn3:=');readln(fn3);
                 write('             fn4:=');readln(fn4);
                 write('             fn5:=');readln(fn5);
                 write('             fn6:=');readln(fn6);
                 write('             fn7:=');readln(fn7);
                 writeln;
                 write('Doresti s† vezi valorile pulsatiilor?
(D/N)');
                 readln(vp);
                 writeln;
                 if (vp='d') or (vp='D') then
                    begin
                         writeln;
                         writeln('Pulsatiile aferente elementelor
modelului sunt:');
                         writeln;
                         omega1:=2*pi*fn1;
                         writeln('     omega1=',omega1:7:2,'
rad/s');
                         omega2:=2*pi*fn2;
                         writeln('     omega2=',omega2:7:2,'
rad/s');
                         omega3:=2*pi*fn3;
                         writeln('     omega3=',omega3:7:2,'
rad/s');
                         omega4:=2*pi*fn4;
                         writeln('     omega4=',omega4:7:2,'
rad/s');
                         omega5:=2*pi*fn5;
                         writeln('     omega5=',omega5:7:2,'
rad/s');
                         omega6:=2*pi*fn6;
                         writeln('     omega6=',omega6:7:2,'
rad/s');
                         omega7:=2*pi*fn7;
                         writeln('     omega7=',omega7:7:2,'
rad/s');
                         readln;
                    end;
             end;
          if (vf<>'d') or (vf<>'D') then
             begin
                   fn1:=3.25;  fn2:=3.25;  fn3:=5;  fn4:=3;
                   fn5:=4.5;  fn6:=5;  fn7:=5;
             end;
     end;

procedure date_init;
```

```
     begin
          m:=80;
          m1:=7.3/100*m;
          m2:=15/100*m;
          m3:=19/100*m;
          m4:=9.7/100*m;
          m5:=36.5/100*m;
          m6:=7/100*m;
          m7:=5.5/100*m;
          fn1:=8;  fn2:=6;  fn3:=8.5;  fn4:=3;
          fn5:=8;  fn6:=5;  fn7:=14;
     end;

procedure calcul_afisare;
     begin
          clrscr;
          textbackground(1);
          textcolor(yellow);
          gotoxy(10,4);
          writeln('P A R A M E T R I I   M  O D E L U L U I  M E C
A N I C:');
          gotoxy(10,5);
writeln('ßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßß'
);
          writeln;
          k12:=m2*sqr(2*pi*fn2)/2;
          k23:=k12;
          k1f:=m1*sqr(2*pi*fn1)-k12;
          k56:=m6*sqr(2*pi*fn6);
          k57:=m7*sqr(2*pi*fn7);
          a:=m3*sqr(2*pi*fn3)-k23;
          b:=m4*sqr(2*pi*fn4);
          c:=m5*sqr(2*pi*fn5)-k56-k57;
          k34:=(a+b-c)/2;
          k35:=(a-b+c)/2;
          k45:=(-a+b+c)/2;
          writeln('  Valorile parametrilor elementelor elastice
sunt:');
          writeln('     k12=',k12:9:2,' N/m');
          writeln('     k23=',k23:9:2,' N/m');
          writeln('     k34=',k34:9:2,' N/m');
          writeln('     k35=',k35:9:2,' N/m');
          writeln('     k45=',k45:9:2,' N/m');
          writeln('     k56=',k56:9:2,' N/m');
          writeln('     k57=',k57:9:2,' N/m');
          writeln('     k1f=',k1f:9:2,' N/m');
          writeln; writeln;
          m:=m1+m2+m3+m4+m5+m6+m7;
          pr1:=m1*100/round(m);
          pr2:=m2*100/round(m);
          pr3:=m3*100/round(m);
          pr4:=m4*100/round(m);
          pr5:=m5*100/round(m);
          pr6:=m6*100/round(m);
          pr7:=m7*100/round(m);
          pr:=pr1+pr2+pr3+pr4+pr5+pr6+pr7;
          writeln('  Masele elementelor componente ale corpului
sunt:');
          writeln('      m1=',m1:9:2,' kg
',pr1:5:2,'%');
          writeln('      m2=',m2:9:2,' kg
',pr2:5:2,'%');
          writeln('      m3=',m3:9:2,' kg
',pr3:5:2,'%');
          writeln('      m4=',m4:9:2,' kg
',pr4:5:2,'%');
          writeln('      m5=',m5:9:2,' kg
',pr5:5:2,'%');
          writeln('      m6=',m6:9:2,' kg
',pr6:5:2,'%');
          writeln('      m7=',m7:9:2,' kg
',pr7:5:2,'%');
          writeln;
          m:=m1+m2+m3+m4+m5+m6+m7;
          writeln('  Masa total† a corpului omenesc este:
m=',round(m),' kg');
          tasta1;
end;

procedure tiparire_imprimanta;
     begin
          clrscr;
          writeln('    Vrei s† tip†resti rezultatele la imprimant†
?');
          writeln;
          writeln('Tasteaz†:');
          writeln;
          writeln('      1 ..... pentru tip†rire la imprimant†');
          writeln;
          writeln('      2 ..... pentru NU');
          writeln;
          readln(vimp);
          writeln;
          if (vimp='1') then
             begin
                clrscr;
                textbackground(7);
                textcolor(5);
                gotoxy(10,10);
                writeln('Esti sigur c† vrei s† tip†resti la
imprimant† valorile');
                write('              parametrilor ? (D/N)');
                readln(cp);
                writeln;
                if (cp='d') or (cp='D') then
                  begin
                     textbackground(5);
                     clrscr;
                     textbackground(1);
                     textcolor(135);
                     gotoxy(30,10);
                     writeln('T I P A R I R E');
                     gotoxy(28,12);
                     writeln('*******************');
                     gotoxy(28,8);
                     writeln('*******************');
                     writeln(lst,'  P A R A M E T R I I   M O D E
L U L U I  M E C A N I C:');
                     writeln(lst,'
ßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßß');
                     writeln(lst,'');
                     writeln(lst,'     k12=',k12:9:2,' N/m');
                     writeln(lst,'     k23=',k23:9:2,' N/m');
                     writeln(lst,'     k34=',k34:9:2,' N/m');
                     writeln(lst,'     k35=',k35:9:2,' N/m');
                     writeln(lst,'     k45=',k45:9:2,' N/m');
                     writeln(lst,'     k56=',k56:9:2,' N/m');
                     writeln(lst,'     k57=',k57:9:2,' N/m');
                     writeln(lst,'     k1f=',k1f:9:2,' N/m');
                     writeln(lst,''); writeln(lst,'');
                     writeln(lst,'  Masele elementelor componente
ale corpului sunt:');
                     writeln(lst,'');
                     writeln(lst,'      m1=',m1:9:2,' kg
',m1*100/round(m):5:1,' %');
                     writeln(lst,'      m2=',m2:9:2,' kg
',m2*100/round(m):5:1,' %');
                     writeln(lst,'      m3=',m3:9:2,' kg
',m3*100/round(m):5:1,' %');
                     writeln(lst,'      m4=',m4:9:2,' kg
',m4*100/round(m):5:1,' %');
                     writeln(lst,'      m5=',m5:9:2,' kg
',m5*100/round(m):5:1,' %');
                     writeln(lst,'      m6=',m6:9:2,' kg
',m6*100/round(m):5:1,' %');
                     writeln(lst,'      m7=',m7:9:2,' kg
',m7*100/round(m):5:1,' %');
                     writeln(lst,'');
                     m:=m1+m2+m3+m4+m5+m6+m7;
                     writeln(lst,'  Masa total† a corpului omenesc
este: m=',round(m),' kg');
                     readln;
                     textcolor(7);
                  end;
             end;
     end;

procedure calcul;
    begin
          m1:=7.3/100*m;
          m2:=15/100*m;
          m3:=19/100*m;
          m4:=9.7/100*m;
          m5:=36.5/100*m;
          m6:=7/100*m;
          m7:=5.5/100*m;
          fn1:=8;  fn2:=6;  fn3:=8.5;  fn4:=3;
          fn5:=8;  fn6:=5;  fn7:=14;
          k12:=m2*sqr(2*pi*fn2)/2;
          k23:=k12;
          k1f:=m1*sqr(2*pi*fn1)-k12;
          k56:=m6*sqr(2*pi*fn6);
          k57:=m7*sqr(2*pi*fn7);
          a:=m3*sqr(2*pi*fn3)-k23;
          b:=m4*sqr(2*pi*fn4);
          c:=m5*sqr(2*pi*fn5)-k56-k57;
          k34:=(a+b-c)/2;
          k35:=(a-b+c)/2;
          k45:=(-a+b+c)/2;
    end;

procedure param;
    begin
         clrscr;
         writeln; writeln; writeln;
         writeln('                Variatia parametrilor kij
functie de masa totala');
         writeln('                --------------------------------
----------------');
         writeln;
         writeln('
==================================================================
=========');
         writeln('   m[kg]  k12[N/m]  k23[N/m]  k34[N/m]  k35[N/m]
k45[N/m]  k56[N/m]  k57[N/m]');
         writeln('         k1f[N/m]');
         writeln('
==================================================================
=========');
         m:=50;
         while m<=100 do
               begin
                  calcul;
                  writeln(' ',m:7:2,' ',k12:9:2,' ',k23:9:2,'
',k34:9:2,' ',k35:9:2,' ',k45:9:2,' ',k56:9:2,' ',k57:9:2);
                  writeln('         ',k1f:9:2);
                  m:=m+6;
               end;
         writeln('
==================================================================
=========');
         readln;
    end;

procedure param_im;
    begin
         writeln(lst);writeln(lst);
         writeln(lst,'            Variatia parametrilor kij
functie de masa totala');
         writeln(lst,'            --------------------------------
----------------');
         writeln(lst);
         writeln(lst,'
==================================================================
=========');
         writeln(lst,'   m[kg]  k12[N/m]  k23[N/m]  k34[N/m]
k35[N/m]  k45[N/m]  k56[N/m]  k57[N/m]');
         writeln(lst,'         k1f[N/m]');
         writeln(lst,'
==================================================================
=========');
         m:=50;
         while m<=100 do
               begin
                  calcul;
                  writeln(lst,' ',m:7:2,' ',k12:9:2,' ',k23:9:2,'
',k34:9:2,' ',k35:9:2,' ',k45:9:2,' ',k56:9:2,' ',k57:9:2);
                  writeln(lst,'         ',k1f:9:2);
                  m:=m+6;
               end;
    end;

procedure param1;
    begin
         clrscr;
         writeln; writeln; writeln;
         writeln('            Variatia maselor partilor corpului
functie de masa totala');
         writeln('            ------------------------------------
---------------------');
         writeln;
         writeln('
==================================================================
=========');
```

```
         writeln('   m[kg]    m1[kg]    m2[kg]    m3[kg]    m4[kg]
m5[kg]    m6[kg]    m7[kg]');
         writeln('
==================================================================
=========');
         m:=50;
         while m<=100 do
               begin
                  calcul;
                  writeln(' ',m:7:2,' ',m1:9:2,' ',m2:9:2,'
',m3:9:2,' ',m4:9:2,' ',m5:9:2,' ',m6:9:2,' ',m7:9:2);
                  m:=m+6;
               end;
         writeln('
==================================================================
=========');
         readln;
    end;

procedure param_im1;
    begin
         writeln(lst);writeln(lst);
         writeln(lst,'         Variatia maselor partilor corpului
functie de masa totala');
         writeln(lst,'         -----------------------------------
----------------------');
         writeln(lst);
         writeln(lst,'   m[kg]    m1[kg]    m2[kg]    m3[kg]
m4[kg]    m5[kg]    m6[kg]    m7[kg]');
         writeln(lst,'
==================================================================
=========');
         m:=50;
         while m<=100 do
               begin
                  calcul;
                  writeln(lst,' ',m:7:2,' ',m1:9:2,' ',m2:9:2,'
',m3:9:2,' ',m4:9:2,' ',m5:9:2,' ',m6:9:2,' ',m7:9:2);
                  m:=m+6;
               end;
    end;

begin
     antet;
     meniu1;
     if (men1=1) then
          date_init;
     if (men1=2) then
          begin
               meniu2;
               if (men2=1) then
```

```
                    masal;
               if (men2=2) then
                    masa2;
          end;
     frecventa;
     calcul_afisare;
     tiparire_imprimanta;
     clrscr;
     writeln;writeln;writeln;writeln;writeln;writeln;
     writeln(' Doriti sa consultati niste date statistice?');
     writeln;
     write('        [d/n]');
     readln(st);
     if (st='d') or (st='D') then
        begin
             param;
             writeln;writeln;
             writeln(' Doriti sa le tipariti la imprimanta?');
             write('        [d/n]');
             readln(im);
             if (im='d') or (im='D') then
                param_im;
             param1;
             writeln;writeln;
             writeln(' Doriti sa le tipariti la imprimanta?');
             write('        [d/n]');
             readln(im);
             if (im='d') or (im='D') then
                param_im1;
        end;
end.
```

ANEXA 4.2

Codul sursă al programului *elastwin.pas* este redat în continuare:

```
program estimare_constante_elastice;
uses wincrt,winprn;
var men1,men2:integer;
    m,s1,m1,m2,m3,m4,m5,m6,m7,r:real;
    fn1,fn2,fn3,fn4,fn5,fn6,fn7,a,b,c:real;
    omega1,omega2,omega3,omega4,omega5,omega6,omega7:real;
    k12,k23,k34,k35,k45,k56,k57,k1f:real;
    ch,vm,vp,vf,vimp,cp,st,im:char;

procedure tasta1;
     begin
          ch:=readkey;
     end;

procedure antet;
     begin
          {textbackground(3);}
          clrscr;
          {textbackground(1);}
          {textcolor(3);}
          gotoxy(20,10);
          writeln('   M  O  D  E  L     M  E  C  A  N  I  C');
          gotoxy(10,12);
          writeln('al corpului omenesc asezat Œn picioare pe o
platformt');
          writeln;
          {delay(2050);}
          clrscr;
          {textbackground(3);
          textcolor(1);}
          gotoxy(12,3);
          writeln('  D E T E R M I N A R E   C O N S T A N T E   E
L A S T I C E');
          writeln;
          writeln;
          tasta1;
     end;

procedure tasta;
     begin
          gotoxy(23,20);
          writeln('>> APASATI ORICE TASTA PENTRU A COMTINUA! <<');
          ch:=readkey;
     end;
```

```
procedure meniu1;
     begin
          writeln('       D A T E   I N I T I A L E');
          writeln('       ~~~~~~~~~~~~~~~~~~~~~~~~~');
          writeln('  Tasteaz†:');
          writeln;
          writeln('     1 ..... pentru folosirea datelor
implicite');
          writeln;
          writeln('     2 ..... pentru a introduce datele de la
tastatur†');
          writeln;
          readln(men1);
          writeln;
          writeln;
     end;

procedure meniu2;
     begin
          writeln('    Tasteaz†:');
          writeln;
          writeln('      1 ..... Dati masa total† a corpului');
          writeln;
          writeln('      2 ..... Dati masele celor 7 elemente
componente');
          writeln;
          readln(men2);
          writeln;
     end;

procedure masa1;
     begin
          clrscr;
          writeln('Dati masa total† a corpului, exprimat† Œn kg
!');
          write('        m=');
          readln(m);
          writeln;
          m1:=7.3/100*m;
          m2:=15/100*m;
          m3:=19/100*m;
          m4:=9.7/100*m;
          m5:=36.5/100*m;
          m6:=7/100*m;
          m7:=5.5/100*m;

          write('Doresti s† vezi valorile maselor componente?
(D/N)');
          readln(vm);
          writeln;
          if (vm='d') or (vm='D') then
```

```
             begin
                  writeln;
                  writeln('Masele elementelor modelului sunt:');
                  writeln;
                  writeln('     m1=',m1:9:3,' kg');
                  writeln('     m2=',m2:9:3,' kg');
                  writeln('     m3=',m3:9:3,' kg');
                  writeln('     m4=',m4:9:3,' kg');
                  writeln('     m5=',m5:9:3,' kg');
                  writeln('     m6=',m6:9:3,' kg');
                  writeln('     m7=',m7:9:3,' kg');
                  tasta;
             end;
     end;

procedure masa2;
     begin
          clrscr;
          writeln('Dati masele elementelor componente ale
corpului, Œn kg !');
          writeln;
          write('      m1=');readln(m1);
          write('      m2=');readln(m2);
          write('      m3=');readln(m3);
          write('      m4=');readln(m4);
          write('      m5=');readln(m5);
          write('      m6=');readln(m6);
          write('      m7=');readln(m7);
          writeln;
          m:=m1+m2+m3+m4+m5+m6+m7;
          writeln('Masa total† a corpului omenesc este:
m=',m:9:3,' kg');
          writeln;
          tasta1;
     end;

procedure frecventa;
     begin
          writeln;
          writeln('Doresti s† dai valorile frecventelor de
rezonant† ale');
          write('    maselor componente ale modelului mecanic ?
(D/N)');
          readln(vf);
          writeln;
          if (vf='d') or (vf='D') then
             begin
                 writeln('Dati frecventele de rezonant†, Œn Hz!');
                 writeln;
                 write('             fn1:=');readln(fn1);
                 write('             fn2:=');readln(fn2);
                 write('             fn3:=');readln(fn3);
                 write('             fn4:=');readln(fn4);
                 write('             fn5:=');readln(fn5);
                 write('             fn6:=');readln(fn6);
                 write('             fn7:=');readln(fn7);
                 writeln;
                 write('Doresti s† vezi valorile pulsatiilor?
(D/N)');
                 readln(vp);
                 writeln;
                 if (vp='d') or (vp='D') then
                    begin
                         writeln;
                         writeln('Pulsatiile aferente elementelor
modelului sunt:');
                         writeln;
                         omega1:=2*pi*fn1;
                         writeln('     omega1=',omega1:7:2,'
rad/s');
                         omega2:=2*pi*fn2;
                         writeln('     omega2=',omega2:7:2,'
rad/s');
                         omega3:=2*pi*fn3;
                         writeln('     omega3=',omega3:7:2,'
rad/s');
                         omega4:=2*pi*fn4;
                         writeln('     omega4=',omega4:7:2,'
rad/s');
                         omega5:=2*pi*fn5;
                         writeln('     omega5=',omega5:7:2,'
rad/s');
                         omega6:=2*pi*fn6;
                         writeln('     omega6=',omega6:7:2,'
rad/s');
                         omega7:=2*pi*fn7;
                         writeln('     omega7=',omega7:7:2,'
rad/s');
                         readln;
                    end;
             end;
          if (vf<>'d') or (vf<>'D') then
             begin
                   fn1:=8;  fn2:=6;  fn3:=8.5;   fn4:=3;
                   fn5:=8;  fn6:=5;  fn7:=14;
             end;
     end;

procedure date_init;
     begin
          m:=80;
          m1:=7.3/100*m;
          m2:=15/100*m;
          m3:=19/100*m;
          m4:=9.7/100*m;
          m5:=36.5/100*m;
          m6:=7/100*m;
          m7:=5.5/100*m;
          fn1:=8;  fn2:=6;  fn3:=8.5;  fn4:=3;
          fn5:=8;  fn6:=5;  fn7:=14;
     end;

procedure calcul_afisare;
     begin
          clrscr;
          {textbackground(1);
          textcolor(yellow);}
          gotoxy(10,4);
          writeln('P A R A M E T R I I   M  O D E L U L U I  M E C
A N I C:');
          gotoxy(10,5);

writeln('ßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßß'
);
          writeln;
          k12:=m2*sqr(2*pi*fn2)/2;
          k23:=k12;
          k1f:=m1*sqr(2*pi*fn1)-k12;
          k56:=m6*sqr(2*pi*fn6);
          k57:=m7*sqr(2*pi*fn7);
          a:=m3*sqr(2*pi*fn3)-k23;
          b:=m4*sqr(2*pi*fn4);
          c:=m5*sqr(2*pi*fn5)-k56-k57;
          k34:=(a+b-c)/2;
          k35:=(a-b+c)/2;
          k45:=(-a+b+c)/2;
          writeln;
          writeln('  Valorile parametrilor elementelor elastice
sunt:');
          writeln('     k12=',k12:9:2,' N/m');
          writeln('     k23=',k23:9:2,' N/m');
          writeln('     k34=',k34:9:2,' N/m');
          writeln('     k35=',k35:9:2,' N/m');
          writeln('     k45=',k45:9:2,' N/m');
          writeln('     k56=',k56:9:2,' N/m');
          writeln('     k57=',k57:9:2,' N/m');
          writeln('     k1f=',k1f:9:2,' N/m');
          writeln; writeln;
          m:=m1+m2+m3+m4+m5+m6+m7;
          writeln('  Masele elementelor componente ale corpului
sunt:');
          writeln('      m1=',m1:9:3,' kg
',m1*100/round(m):5:1,' %');
```

```
          writeln('      m2=',m2:9:3,' kg
',m2*100/round(m):5:1,' %');
          writeln('      m3=',m3:9:3,' kg
',m3*100/round(m):5:1,' %');
          writeln('      m4=',m4:9:3,' kg
',m4*100/round(m):5:1,' %');
          writeln('      m5=',m5:9:3,' kg
',m5*100/round(m):5:1,' %');
          writeln('      m6=',m6:9:3,' kg
',m6*100/round(m):5:1,' %');
          writeln('      m7=',m7:9:3,' kg
',m7*100/round(m):5:1,' %');
          writeln;
          m:=m1+m2+m3+m4+m5+m6+m7;
          writeln('  Masa total† a corpului omenesc este:
m=',round(m),' kg');
          tastal;
end;

procedure calcul;
    begin
            m1:=7.3/100*m;
            m2:=15/100*m;
            m3:=19/100*m;
            m4:=9.7/100*m;
            m5:=36.5/100*m;
            m6:=7/100*m;
            m7:=5.5/100*m;
            fn1:=8;  fn2:=6;  fn3:=8.5;  fn4:=3;
            fn5:=8;  fn6:=5;  fn7:=14;
            k12:=m2*sqr(2*pi*fn2)/2;
            k23:=k12;
            k1f:=m1*sqr(2*pi*fn1)-k12;
            k56:=m6*sqr(2*pi*fn6);
            k57:=m7*sqr(2*pi*fn7);
            a:=m3*sqr(2*pi*fn3)-k23;
            b:=m4*sqr(2*pi*fn4);
            c:=m5*sqr(2*pi*fn5)-k56-k57;
            k34:=(a+b-c)/2;
            k35:=(a-b+c)/2;
            k45:=(-a+b+c)/2;
    end;

procedure param;
    begin
         clrscr;
         writeln; writeln; writeln;
         writeln('               Variatia parametrilor kij
functie de masa totala');
         writeln('               --------------------------------
-----------------');
```

```
         writeln;
         writeln('
==================================================================
=========');
         writeln('   m[kg]  k12[N/m]  k23[N/m]  k34[N/m]  k35[N/m]
k45[N/m]  k56[N/m]  k57[N/m]');
         writeln('        k1f[N/m]');
         writeln('
==================================================================
=========');
         m:=50;
         while m<=100 do
               begin
                  calcul;
                  writeln(' ',m:7:2,' ',k12:9:2,' ',k23:9:2,'
',k34:9:2,' ',k35:9:2,' ',k45:9:2,' ',k56:9:2,' ',k57:9:2);
                  writeln('       ',k1f:9:2);
                  m:=m+6;
               end;
         writeln('
==================================================================
=========');
         readln;
    end;

procedure param1;
    begin
         clrscr;
         writeln; writeln; writeln;
         writeln('            Variatia maselor partilor corpului
functie de masa totala');
         writeln('            ------------------------------------
---------------------');
         writeln;
         writeln('
==================================================================
=========');
         writeln('   m[kg]    m1[kg]    m2[kg]    m3[kg]    m4[kg]
m5[kg]    m6[kg]    m7[kg]');
         writeln('
==================================================================
=========');
         m:=50;
         while m<=100 do
               begin
                  calcul;
                  writeln(' ',m:7:2,' ',m1:9:2,' ',m2:9:2,'
',m3:9:2,' ',m4:9:2,' ',m5:9:2,' ',m6:9:2,' ',m7:9:2);
                  m:=m+6;
               end;
         writeln('
==================================================================
=========');
         readln;
    end;

begin
     antet;
     meniu1;
     if (men1=1) then
          date_init;
     if (men1=2) then
          begin
               meniu2;
               if (men2=1) then
                    masa1;
               if (men2=2) then
                    masa2;
          end;
     frecventa;
     calcul_afisare;
     clrscr;
     writeln;writeln;writeln;writeln;writeln;writeln;
     writeln(' Doriti sa consultati niste date statistice?');
     writeln;
     write('        [d/n]');
     readln(st);
     if (st='d') or (st='D') then
        begin
             param;
             writeln;writeln;
             writeln(' Doriti sa le tipariti la imprimanta?');
             write('        [d/n]');
             readln(im);
             param1;
             writeln;writeln;
             writeln(' Doriti sa le tipariti la imprimanta?');
             write('        [d/n]');
             readln(im);
       end;
end.
```

ANEXA 4.3

Rezultatele rulării programului *elastfin.pas*:

```
P A R A M E T R I I   M O D E L U L U I   M E C A N I C:


   k12=  8527.34 N/m
   k23=  8527.34 N/m
   k34=  1690.47 N/m
   k35= 33137.39 N/m
   k45=  1066.71 N/m
   k56=  5526.98 N/m
   k57= 34046.19 N/m
   k1f=  6228.12 N/m


Masele elementelor componente ale corpului sunt:

    m1=     5.84 kg                        7.3 %
    m2=    12.00 kg                       15.0 %
    m3=    15.20 kg                       19.0 %
    m4=     7.76 kg                        9.7 %
    m5=    29.20 kg                       36.5 %
    m6=     5.60 kg                        7.0 %
    m7=     4.40 kg                        5.5 %

Masa totală a corpului omenesc este: m=80 kg


          Variatia parametrilor kij functie de masa totala
          ------------------------------------------------

 ==========================================================================
 m[kg]  k12[N/m]  k23[N/m]  k34[N/m]  k35[N/m]  k45[N/m]  k56[N/m]  k57[N/m
       k1f[N/m]
 ==========================================================================
 50.00   5329.59   5329.59   1056.54  20710.87    666.69   3454.36  21278.8
         3892.57
 56.00   5969.14   5969.14   1183.33  23196.18    746.69   3868.88  23832.3
         4359.68
 62.00   6608.69   6608.69   1310.11  25681.48    826.70   4283.41  26385.8
         4826.79
 68.00   7248.24   7248.24   1436.90  28166.79    906.70   4697.93  28939.2
         5293.90
 74.00   7887.79   7887.79   1563.68  30652.09    986.70   5112.46  31492.7
         5761.01
 80.00   8527.34   8527.34   1690.47  33137.39   1066.71   5526.98  34046.1
         6228.12
 86.00   9166.89   9166.89   1817.25  35622.70   1146.71   5941.50  36599.6
         6695.22
 92.00   9806.44   9806.44   1944.04  38108.00   1226.71   6356.03  39153.1
         7162.33
 98.00  10445.99  10445.99   2070.82  40593.31   1306.72   6770.55  41706.5
         7629.44
```

Var maselor partilor corpului functie de masa totala

m[kg]	m2[kg]	m3[kg]	m4[kg]	m5[kg]	m6[kg]	m7[kg
50.00	7.50	9.50	4.85	18.25	3.50	2.7
56.00	8.40	10.64	5.43	20.44	3.92	3.0
62.00	9.30	11.78	6.01	22.63	4.34	3.4
68.00	10.20	12.92	6.60	24.82	4.76	3.7
74.00	1.10	14.06	7.18	27.01	5.18	4.0
80.00	2.00	15.20	7.76	29.20	5.60	4.4
86.00	.90	16.34	8.34	31.39	6.02	4.7
92.00	80	17.48	8.92	33.58	6.44	5.0
98.00	70	18.62	9.51	35.77	6.86	5.3

ANEXA 4.4

Codul sursă al acestui programului *amortizp.pas* este redat în continuare:

```
program estimare_constante_elastice;
uses crt,printer;
var men1,men2:integer;
    m,s1,m1,m2,m3,m4,m5,m6,m7,r:real;
    fn1,fn2,fn3,fn4,fn5,fn6,fn7,a,b,c:real;
    omega1,omega2,omega3,omega4,omega5,omega6,omega7:real;
    k12,k23,k34,k35,k45,k56,k57,k1f:real;
    p1,p2,p3,p4,p5,p6,p7:real;
    n13,n34,n35,n45,n56,n57,c13,c34,c45,c56,c57:real;
    c13cr,c34cr,c45cr,c56cr,c57cr,c45cr1:real;
    a3,b3,c3,v3,a4,b4,c4,v4,a5,b5,c5,v5:real;
    ch,vm,vp,vf,vimp,cp,ir:char;

procedure tasta1;
     begin
          ch:=readkey;
     end;

procedure antet;
     begin
          textbackground(3);
          clrscr;
          textbackground(1);
          textcolor(3);
          gotoxy(20,10);
          writeln('  M  O  D  E  L    M  E  C  A  N  I  C');
          gotoxy(10,12);
    writeln('al corpului omenesc asezat Œn picioare pe o platform†
vibrant†');
          writeln;
          delay(3050);
          clrscr;
          textbackground(3);
          textcolor(1);
          gotoxy(12,3);
          writeln('  E S T I M A R E   P A R A M E T R I   M O D
E L');
          writeln;
          writeln;
          tasta1;
     end;

procedure tasta;
     begin
          gotoxy(23,20);
          writeln('>> APASATI ORICE TASTA PENTRU A COMTINUA! <<');
          ch:=readkey;
     end;

procedure meniu1;
     begin
          writeln('       D A T E   I N I T I A L E');
          writeln('       ~~~~~~~~~~~~~~~~~~~~~~~~~');
          writeln('  Tasteaz†:');
          writeln;
          writeln('           1  .....  pentru  folosirea  datelor
implicite');
          writeln;
          writeln('       2 ..... pentru a introduce datele de la
tastatur†');
          writeln;
          readln(men1);
          writeln;
          writeln;
     end;

procedure meniu2;
     begin
          writeln('    Tasteaz†:');
          writeln;
          writeln('      1 ..... Dati masa total† a corpului');
          writeln;
          writeln('         2  .....  Dati  masele  celor  7  elemente
componente');
          writeln;
          readln(men2);
          writeln;
     end;

procedure masa1;
     begin
          clrscr;
          writeln;writeln;
          writeln;writeln;
          writeln('   Dati masa total† a corpului, exprimat† Œn kg
!');
          writeln;
          write('          m=');
          readln(m);
          m1:=7.3/100*m;
          m2:=15/100*m;
          m3:=19/100*m;
          m4:=9.7/100*m;
          m5:=36.5/100*m;
          m6:=7/100*m;
          m7:=5.5/100*m;
```

```
          writeln;
          write('Doresti   s†   vezi   valorile   maselor   componente?
(D/N)');
          readln(vm);
          writeln;
          if (vm='d') or (vm='D') then
             begin
                  writeln;
                  writeln('Masele elementelor modelului sunt:');
                  writeln;
                  writeln('     m1=',m1:9:3,' kg');
                  writeln('     m2=',m2:9:3,' kg');
                  writeln('     m3=',m3:9:3,' kg');
                  writeln('     m4=',m4:9:3,' kg');
                  writeln('     m5=',m5:9:3,' kg');
                  writeln('     m6=',m6:9:3,' kg');
                  writeln('     m7=',m7:9:3,' kg');
                  tasta;
             end;
     end;

procedure masa2;
     begin
          clrscr;
          writeln('Dati   masele   elementelor   componente   ale
corpului, Œn kg !');
          writeln;
          write('      m1=');readln(m1);
          write('      m2=');readln(m2);
          write('      m3=');readln(m3);
          write('      m4=');readln(m4);
          write('      m5=');readln(m5);
          write('      m6=');readln(m6);
          write('      m7=');readln(m7);
          writeln;
          m:=m1+m2+m3+m4+m5+m6+m7;
          delay(850);
          writeln('Masa   total†   a   corpului   omenesc   este:
m=',m:9:2,' kg');
          writeln;
          tasta1;
     end;

procedure amortizare;
     begin
          clrscr;
          writeln;writeln;writeln;
       writeln('Dati  valoarea  procentual†  a  raportului  dintre
amortizarea');
```

```
       writeln('  real† si cea critic† pentru fiecare element cu
amortizare');
          writeln('  vƒscoas† Œn parte !');
          writeln;
          write('     n13=');readln(n13);
          write('     n34=');readln(n34);
          write('     n45=');readln(n45);
          write('     n56=');readln(n56);
          write('     n57=');readln(n57);
          writeln;
          n13:=n13/100;n34:=n34/100;
          n45:=n45/100;n56:=n56/100;n57:=n57/100;
          tastal;
     end;

procedure frecventa;
     begin
          writeln;
          writeln('Doresti s† dai valorile frecventelor de
rezonant† ale');
          write('     maselor componente ale modelului mecanic ?
(D/N)');
          readln(vf);
          writeln;
          if (vf='d') or (vf='D') then
             begin
                 writeln('Dati frecventele de rezonant†, Œn Hz!');
                 writeln;
                 write('             fn1=');readln(fn1);
                 write('             fn2=');readln(fn2);
                 write('             fn3=');readln(fn3);
                 write('             fn4=');readln(fn4);
                 write('             fn5=');readln(fn5);
                 write('             fn6=');readln(fn6);
                 write('             fn7=');readln(fn7);
                 writeln;
                 p1:=2*pi*fn1; p2:=2*pi*fn2; p3:=2*pi*fn3;
                 p4:=2*pi*fn4; p5:=2*pi*fn5; p6:=2*pi*fn6;
                 p7:=2*pi*fn7;
                 write('Doresti s† vezi valorile pulsatiilor?
(D/N)');
                 readln(vp);
                 writeln;
                 if (vp='d') or (vp='D') then
                    begin
                         writeln;
                   writeln('Pulsatiile aferente elementelor
modelului sunt:');
                         writeln;
                         writeln('    p1=',p1:7:2,' rad/s');
                         writeln('    p2=',p2:7:2,' rad/s');
                         writeln('     p3=',p3:7:2,' rad/s');
                         writeln('     p4=',p4:7:2,' rad/s');
                         writeln('     p5=',p5:7:2,' rad/s');
                         writeln('     p6=',p6:7:2,' rad/s');
                         writeln('     p7=',p7:7:2,' rad/s');
                         readln;
                    end;
             end;
          if (vf<>'d') or (vf<>'D') then
             begin
                   fn1:=8;  fn2:=6;  fn3:=8.5;  fn4:=3;
                   fn5:=8;  fn6:=5;  fn7:=14;
                   p1:=2*pi*fn1; p2:=2*pi*fn2; p3:=2*pi*fn3;
                   p4:=2*pi*fn4; p5:=2*pi*fn5; p6:=2*pi*fn6;
                   p7:=2*pi*fn7;
              end;
     end;

procedure date_init;
     begin
          m:=80;
          m1:=7.3/100*m;
          m2:=15/100*m;
          m3:=19/100*m;
          m4:=9.7/100*m;
          m5:=36.5/100*m;
          m6:=7/100*m;
          m7:=5.5/100*m;
          fn1:=8;  fn2:=6;  fn3:=8.5;  fn4:=3;
          fn5:=8;  fn6:=5;  fn7:=14;
          p1:=2*pi*fn1;p2:=2*pi*fn2;p3:=2*pi*fn3;p4:=2*pi*fn4;
          p5:=2*pi*fn5;p6:=2*pi*fn6;p7:=2*pi*fn7;
          n13:=0.05;n34:=0.10;n45:=0.25;n56:=0.15;n57:=0.10;
     end;

procedure calcul;
     begin
          k12:=m2*p2*p2/2;
          k23:=k12;
          k1f:=m1*p1*p1/(1-n13*n13)-k12;
          c13cr:=2*sqrt((k12+k1f)*m1);
          c13:=n13*c13cr;
          a3:=1-sqr(n34);
          b3:=2*n34*(n34-n13)*sqrt(m1*k12);
          c3:=-(sqr(m3*p3)+sqr(n13-n34)*m1*k12);
          v3:=(-b3+sqrt(b3*b3-4*a3*c3))/(2*a3);
          a:=v3*v3/m3-k23;
          c34cr:=2*(sqrt(m3*(k23+a))-sqrt(m1*k12));
          c34:=n34*c34cr;
          a4:=1-sqr(n45);
          b4:=n45*(n45-n34)*c34cr;
          c4:=-sqr(m4*p4)-sqr(n45-n34)*sqr(c34cr)/4;
          v4:=(-b4+sqrt(b4*b4-4*a4*c4))/(2*a4);
          b:=v4*v4/m4;
          c45cr:=-(2*sqrt(m4*b)-c34cr);
          c45:=n45*c45cr;
          k56:=m6*p6*p6/(1-sqr(n56));
          c56cr:=2*sqrt(k56*m6);
          c56:=n56*c56cr;
          k57:=m7*p7*p7/(1-sqr(n57));
          c57cr:=2*sqrt(k57*m7);
          c57:=n57*c57cr;
          c:=m5*sqr(p5)+(c45+c56+c57)/(4*m5)-k56-k57;
          k34:=(a+b-c)/2;
          k35:=(a-b+c)/2;
          k45:=(-a+b+c)/2;
          c45cr1:=2*sqrt(m5*sqr(k35+k45+k56+k57)-c56cr-c57cr);
          readln;
     end;

procedure afisare;
     begin
          clrscr;
          textbackground(1);
          textcolor(yellow);
          gotoxy(10,4);
          writeln('P A R A M E T R I I   M O D E L U L U I  M E C
A N I C:');
          gotoxy(10,5);

writeln('ßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßß'
);
          writeln;
          writeln;
          writeln('     k12=',k12:9:2);
          writeln('     k23=',k23:9:2);
          writeln('     k34=',k34:9:2);
          writeln('     k35=',k35:9:2);
          writeln('     k45=',k45:9:2);
          writeln('     k56=',k56:9:2);
          writeln('     k57=',k57:9:2);
          writeln('     k1f=',k1f:9:2);
          writeln;
          writeln('     c13=',c13:9:2);
          writeln('     c34=',c34:9:2);
          writeln('     c45=',c45:9:2);
          writeln('     c56=',c56:9:2);
          writeln('     c57=',c57:9:2);
          readln;
     end;

procedure tiparire_imprimanta;
```

```
     begin
          clrscr;
          writeln('    Vrei s† tip†resti rezultatele la imprimant†
?');
          writeln;
          writeln('Tasteaz†:');
          writeln;
          writeln('      1 ..... pentru tip†rire la imprimant†');
          writeln;
          writeln('      2 ..... pentru NU');
          writeln;
          readln(vimp);
          writeln;
          if (vimp='1') then
             begin
                clrscr;
                textbackground(7);
                textcolor(5);
                gotoxy(10,10);
           writeln('Esti sigur c† vrei s† tip†resti la imprimant†
valorile');
                write('              parametrilor ? (D/N)');
                readln(cp);
                writeln;
                if (cp='d') or (cp='D') then
                  begin
                     writeln('Doresti     s†     tip†resti     la
imprimant†:');
                     writeln;
                     writeln('     1 ..... numai rezultatele');
                     writeln;
                     writeln('     2 ..... si datele de intrare');
                     writeln;
                     readln(ir);
                     textbackground(5);
                     clrscr;
                     textbackground(1);
                     textcolor(135);
                     gotoxy(30,10);
                     writeln('T I P A R I R E');
                     gotoxy(30,12);
                     writeln('***************');
                     if ir='2' then begin
                        writeln(lst,'Datele de intrare!');
                        writeln(lst,'~~~~~~~~~~~~~~~~~~');
                        writeln(lst,'');
                        writeln(lst,'   fn1=',fn1:7:2,' Hz');
                        writeln(lst,'   fn2=',fn2:7:2,' Hz');
                        writeln(lst,'   fn3=',fn3:7:2,' Hz');
                        writeln(lst,'   fn4=',fn4:7:2,' Hz');
                        writeln(lst,'   fn5=',fn5:7:2,' Hz');
```

```
                        writeln(lst,'   fn6=',fn6:7:2,' Hz');
                        writeln(lst,'   fn7=',fn7:7:2,' Hz');
                        writeln(lst,'');writeln(lst,'');
                        writeln(lst,'   n13=',n13*100:6:2,' %');
                        writeln(lst,'   n34=',n34*100:6:2,' %');
                        writeln(lst,'   n45=',n45*100:6:2,' %');
                        writeln(lst,'   n56=',n56*100:6:2,' %');
                        writeln(lst,'   n57=',n57*100:6:2,' %');
                        writeln(lst,'');writeln(lst,'');
                     end;
      writeln(lst,'P A R A M E T R I I   M O D E L U L U I  M E C
A N I C:');

writeln(lst,'ßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßß
ßßß');
                     writeln(lst,'');
                     writeln(lst,'    k12=',k12:9:2);
                     writeln(lst,'    k23=',k23:9:2);
                     writeln(lst,'    k34=',k34:9:2);
                     writeln(lst,'    k35=',k35:9:2);
                     writeln(lst,'    k45=',k45:9:2);
                     writeln(lst,'    k56=',k56:9:2);
                     writeln(lst,'    k57=',k57:9:2);
                     writeln(lst,'    k1f=',k1f:9:2);
                     writeln(lst,''); writeln(lst,'');
             writeln(lst,'Masele   elementelor   componente   ale
corpului sunt:');

writeln(lst,'************************************************');
                     writeln(lst,'');}
                     writeln(lst,'     m1=',m1:9:3,' kg');
                     writeln(lst,'     m2=',m2:9:3,' kg');
                     writeln(lst,'     m3=',m3:9:3,' kg');
                     writeln(lst,'     m4=',m4:9:3,' kg');
                     writeln(lst,'     m5=',m5:9:3,' kg');
                     writeln(lst,'     m6=',m6:9:3,' kg');
                     writeln(lst,'     m7=',m7:9:3,' kg');
                     writeln(lst,'');
                     m:=m1+m2+m3+m4+m5+m6+m7;
           writeln(lst,'Masa   total†   a   corpului   omenesc   este:
m=',m:7:3,' kg');
                     writeln(lst,'');writeln(lst,'');
                     writeln(lst,'    c13=',c13:9:2);
                     writeln(lst,'    c34=',c34:9:2);
                     writeln(lst,'    c45=',c45:9:2);
                     writeln(lst,'    c56=',c56:9:2);
                     writeln(lst,'    c57=',c57:9:2);
                     writeln;
                  end;
             end;
     end;
```

```
begin
     antet;
     meniu1;
     if (men1=1) then
          date_init;
     if (men1=2) then
          begin
               meniu2;
               if (men2=1) then
                    masa1;
               if (men2=2) then
                    masa2;
               amortizare;
          end;
     frecventa;
     calcul;
     afisare;
     tiparire_imprimanta;
end.
```

ANEXA 4.5

Codul sursă al acestui programului *parametp.bat* este redat în continuare:

```
ECHO OFF

:RELUARE
CLS
ECHO
ECHO -------------------------------------------------------------
-----
ECHO !               K : fara amortizare
ECHO !               C : cu amortizare
ECHO !               X : EXIT
ECHO -------------------------------------------------------------
-----
ECHO
CHOICE /c:KCX Alegeti o comanda
IF ERRORLEVEL 3 GOTO FINAL
IF ERRORLEVEL 2 GOTO AM
IF ERRORLEVEL 1 GOTO ARC

:ARC
ECHO -------------------------------------------------------------
-----
ECHO !              A: programul e pe discheta
ECHO !              C: programul e pe harddisc
ECHO -------------------------------------------------------------
-----
ECHO
CHOICE /c:AC Optiunea:
IF ERRORLEVEL 2 GOTO ARC1
IF ERRORLEVEL 1 GOTO ARC2

:ARC2
CALL A:\ELASTFIN.EXE
PAUSE
GOTO RELUARE

:ARC1
CALL C:\LUCRU\ELASTFIN.EXE
PAUSE
GOTO RELUARE

:AM
ECHO -------------------------------------------------------------
-----
ECHO !              A: programul e pe discheta
ECHO !              C: programul e pe harddisk
ECHO -------------------------------------------------------------
-----
ECHO
CHOICE /c:AC Optiunea:
IF ERRORLEVEL 2 GOTO AM1
IF ERRORLEVEL 1 GOTO AM2

:AM2
CALL A:\AMORTIZP.EXE
GOTO RELUARE

:AM1
CALL C:\LUCRU\AMORTIZP.EXE
GOTO RELUARE

:FINAL
ECHO
******************************************************************
ECHO !
ECHO !     Daca nu a rulat programul e posibil ca sa nu fi fost
gasit
ECHO !
ECHO !     Deci:  - deschide fisierul paramet.bat pentru editare
ECHO !               (in NortonCommander cu <F4>), sau folosind
ECHO !               editorul Notepad (in Windows)
ECHO !
ECHO !            - modifica in mod corect calea la care se
gaseste
ECHO !               fisierul elastfin.exe si/sau fisierul
amortiz.exe
ECHO !
ECHO !            - salveaza modificarile aduse (in NC cu <F2>) si
ECHO !           inchide fisierul (in NC cu <F10>)
ECHO !
ECHO
******************************************************************
```

ECHO OFF

ANEXA 4.6

Codul sursă al acestui programului *param.pas* este redat în continuare:

```
program estimare_parametri_model;
uses crt,printer;
var fn,k,c,omega_n,m,p,n,ccr,delta,tau,csi,eta,omega:real;
    ch,men,tv,im,list,rul:char;
    rulo:byte;

procedure antet;
     begin
          textbackground(1);
          clrscr;
          textbackground(3);
          textcolor(1);
          gotoxy(20,10);
          writeln('   M   O   D   E   L     M   E   C   A   N   I   C');
          gotoxy(10,12);
          writeln('al corpului omenesc asezat Œn picioare pe o
platformt vibrantt');
          gotoxy(16,14);
          writeln(' - cazul vibratiilor cu frecvente de valori
mici - ');
          writeln;
          delay(2500);
          clrscr;
          textbackground(1);
          textcolor(3);
          gotoxy(18,3);
          writeln('  D E T E R M I N A R E    P A R A M E T R I
');
          gotoxy(32,5);
          writeln('   M O D E L   ');
          writeln;
          readln;
     end;

procedure tasta;
     begin
          gotoxy(23,23);
          writeln('>> APASATI ORICE TASTA PENTRU A COMTINUA! <<');
          ch:=readkey;
     end;
```

```
procedure meniu;
     begin
          writeln('       D A T E   I N I T I A L E');
          writeln('       ~~~~~~~~~~~~~~~~~~~~~~~~~');
          writeln('  Tasteaz†:');
          writeln;
          writeln('     1 ..... pentru folosirea datelor
implicite');
          writeln;
          writeln('     2 ..... pentru a introduce datele de la
tastatur†');
          writeln;
          readln(men);
          writeln;
          writeln;
     end;

procedure tip_vibratie;
     begin
          clrscr;
          gotoxy(10,10);
          writeln('T I P    V I B R A T I I');
          gotoxy(10,11);
          writeln('~~~~~~~~~~~~~~~~~~~~~~~~');
          writeln;
          writeln('          Tasteaz†:');
          writeln;
          writeln('      1 ..... pentru vibratii "ne"amortizate');
          writeln;
          writeln('      2 ..... pentru vibratii amortizate');
          writeln;
          readln(tv);
          writeln;
     end;

procedure listare;
 begin
   clrscr;
   m:=40;
   writeln; writeln;
   writeln(' fn=    2,5         2,75         3          3,25
3,5    Hz');
   writeln;
   repeat
       fn:=2.5;
       writeln(' m=',m:5:1,' kg');
       repeat
             omega_n:=2*pi*fn;
             k:=m*sqr(omega_n);
             write('k=',k:10:3,'  ');
             fn:=fn+0.25;
```

```
       until fn>3.5;
       if m=80 then
          readln;
       writeln;
       m:=m+5;
   until m=110;
   tasta;
 end;

procedure imprimanta;
 begin
   clrscr;
   m:=40;
   writeln(lst);writeln(lst);
   writeln(lst,' fn=    2,5         2,75         3          3,25
3,5    Hz');
   writeln(lst);
   gotoxy(20,10);
   writeln('Asteapta!  Se tipareste!');
   repeat
       fn:=2.5;
       writeln(lst,' m=',m:5:1,' kg');
       repeat
             omega_n:=2*pi*fn;
             k:=m*sqr(omega_n);
             write(lst,'k=',k:10:3,'  ');
             fn:=fn+0.25;
       until fn>3.5;
       writeln(lst);
       m:=m+5;
   until m=110;
   readln;
 end;

procedure listare_k_am;
 begin
   clrscr;
   m:=40;
   if (men='1') then
      n:=60;
   if (men='2') then
      begin
           writeln('Dati valoarea procentajului amortiz†rii in %!');
           write('              n=');
           readln(n);
      end;
   writeln('fn=    2,5        2,75        3         3,25         3,5
Hz');
   writeln;
   repeat
       fn:=2.5;
       writeln('m=',m:5:1,' kg');
       repeat
             p:=2*pi*fn;
             k:=m*p*p/(1-(n*n/10000));
             write('k=',k:9:2,'  ');
             fn:=fn+0.25;
       until fn>3.5;
       fn:=2.5;
       writeln;
       repeat
             p:=2*pi*fn;
             k:=m*p*p/(1-(n*n/10000));
             c:=(n/100)*2*sqrt(m*k);
             ccr:=2*sqrt(m*k);
             write('c=',c:9:2,'  ');
             fn:=fn+0.25;
       until fn>3.5;
       writeln;
       m:=m+10;
   until m=110;
   readln;
 end;

procedure imprimanta_k_am;
 begin
   clrscr;
   m:=40;
   writeln(lst);writeln(lst);writeln(lst);writeln(lst);
   writeln(lst,'fn=      2,5         2,75         3          3,25
3,5    Hz');
   writeln(lst);
   gotoxy(20,10);
   writeln('Asteapta!  Se tipareste!');
   repeat
       fn:=2.5;
       writeln(lst,'m=',m:5:1,' kg');
       repeat
             p:=2*pi*fn;
             k:=m*p*p/(1-(n*n/10000));
             write(lst,'k=',k:10:2,'  ');
             fn:=fn+0.25;
       until fn>3.5;
       fn:=2.5;
       writeln(lst);
       repeat
             p:=2*pi*fn;
             k:=m*p*p/(1-(n*n/10000));
             c:=(n/100)*2*sqrt(m*k);
             ccr:=2*sqrt(m*k);
             write(lst,'c=',c:10:2,'  ');
             fn:=fn+0.25;
       until fn>3.5;
       writeln(lst);
       m:=m+10;
   until m=110;
   readln;
 end;

procedure listare_k_am_n;
 begin
   clrscr;
   m:=75;
   if men='2' then
      begin
         writeln;
         writeln('Dati frecventa proprie, Śn Hz !');
         write('      fn=');
         readln(fn);
         writeln;
      end
   else
       fn:=3.25;
   writeln;
   writeln('Variatie parametri model functie de raportul n=c/ccr');
   writeln('-----------------------------------------------------');
   writeln('     n[%]      k[N/m]     c[Ns/m]   ccr[Ns/m]     n ');
   writeln('-----------------------------------------------------');
   n:=0.00001;
   repeat
         p:=2*pi*fn;
         k:=m*p*p/(1-(n*n/10000));
         c:=2*sqrt(m*k)*n/100;
         ccr:=2*sqrt(m*k);
         writeln('    ',n:5:1,'    ',k:9:2,'    ',c:7:2,'
',ccr:7:2,'    ',c/ccr:5:3);
         n:=n+5-0.00001;
   until n>100;
   readln;
   writeln('m=',m:5:1,' kg');
   readln;
 end;

procedure imprimanta_k_am_n;
 begin
   writeln(lst); writeln(lst);
   writeln(lst,'Variatie parametri model functie de raportul
c/ccr');
   writeln(lst,'---------------------------------------------------
-');
   writeln(lst,'     n[%]      k[N/m]     c[Ns/m]   ccr[Ns/m]     n
');
```

```
   writeln(lst,'----------------------------------------------------
-');
   n:=0.00001;
   repeat
         p:=2*pi*fn;
         k:=m*p*p/(1-(n*n/10000));
         c:=2*sqrt(m*k)*n/100;
         ccr:=2*sqrt(m*k);
         writeln(lst,'    ',n:5:1,'    ',k:9:2,'    ',c:7:2,'
',ccr:7:2,'    ',c/ccr:5:3);
         n:=n+5-0.00001;
   until n>100;
   writeln(lst,'m=',m:5:1,' kg');
 end;

begin
  clrscr;
  rulo:=1;
  antet;
  while rulo=1 do begin
     meniu;
     if (men='1') then
        begin
             fn:=3.25;
             m:=65;
        end;
     if (men='2') then
        begin
             writeln('Dati masa modelului, Śn kg !');
             write('      m=');
             readln(m);
             writeln;
             writeln('Dati frecventa proprie, Śn Hz !');
             write('      fn=');
             readln(fn);
             writeln;
        end;
     tip_vibratie;
     if tv='1' then
        begin
            omega_n:=2*pi*fn;
            k:=m*sqr(omega_n);
            clrscr;
            gotoxy(10,8);
            writeln('Parametrii modelului sunt:');
            gotoxy(20,10);
            writeln('k=',k:10:3,' N/m');
            gotoxy(20,12);
            writeln('m=',m:6:2,' kg');
            readln;
        end;
     if tv='2' then
        begin
             p:=2*pi*fn;
             if (men='1') then
                n:=60;
             if (men='2') then
                begin
                     writeln('Dati valoarea procentajului
amortiz†rii in %!');
                     write('              n=');
                     readln(n);
                end;
             k:=m*p*p/(1-(n*n/10000));
             c:=(n/100)*2*sqrt(m*k);
             ccr:=2*sqrt(m*k);
             delta:=2*pi*n/100;
             clrscr;
             gotoxy(10,8);
             writeln('Parametrii modelului sunt:');
             gotoxy(20,10);
             k:=k/1000;
             writeln('k=',k:7:3,' KN/m');
             gotoxy(20,12);
             writeln('m=',m:7:2,' kg');
             gotoxy(20,14);
             c:=c/1000;
             writeln('c=',c:7:3,' KNs/m');
             writeln;
             writeln('Doresti sa cunosti si coef.de amortizare
critic† ? [d/n]');
             if (readkey='d') or (readkey='D') then
                begin
                     writeln;
                     ccr:=ccr/1000;
                     writeln('    c_cr=',ccr:10:3,' KNs/m');
             end;
             readln;
             writeln;
             writeln('Vibratiile sunt libere ? [d/n]');
             if (readkey='n') or (readkey='N') then
                begin
                     if men='2' then
                     begin
                          writeln;
                          writeln('Dati pulsatia vibratiei fortate
care Śntretine miscarea!');
                          write('     omega:=');
                          readln(omega);
                     end;
                     if men='1' then omega:=2*pi*1.5;
                     csi:=c/(2*sqrt(k*m));
                     eta:=omega/p;
                     tau:=sqrt((1+sqr(2*csi)*sqr(eta))/(sqr(1-
sqr(eta))+2*sqr(csi)*sqr(eta)));
                     writeln;
                     writeln;
                     writeln('Factorul de amortizare este:
csi=',2*csi:6:3);
                     writeln;
                     writeln('Transmisibilitatea este:
tau=',tau:6:3);
                     writeln;
                     writeln('Pulsatia relativ† este:
eta=',eta:6:3);
                     readln;
                end;
        end;
   if tv='1' then
     begin
        writeln('Esti interesat de niste date statistice?');
        write('            (d/n)');
        readln(list);
           if (list='d') or (list='D') then
             begin
                listare;
                writeln('Doresti sa fie listate la imprimanta
rezultate statistice?');
                write('           (d/n)');
                readln(im);
                if (im='d') or (im='D') then
                   imprimanta;
             end;
     end;
  if tv='2' then
     begin
        writeln('Esti interesat de niste date statistice?');
        write('            (d/n)');
        readln(list);
           if (list='d') or (list='D') then
             begin
                listare_k_am;
                writeln('Doresti sa fie listate la imprimanta
rezultate statistice?');
                write('           (d/n)');
                readln(im);
                if (im='d') or (im='D') then
                   begin
                        imprimanta_k_am;
                   end;
             end;
     end;
  if tv='2' then
     begin
        writeln('Doriti sa continuati?');
        write('            [d/n]');
        readln(list);
           if (list='d') or (list='D') then
             begin
                listare_k_am_n;
                writeln('Doresti sa fie listarea la imprimanta?');
                write('            [d/n]');
                readln(im);
                if (im='d') or (im='D') then
                   begin
                        imprimanta_k_am_n;
                   end;
             end;
     end;
     writeln;
     writeln('Mai doriti sa rulati o data programul?');
     write('         [d/n]');
     readln(rul);
     if rul='d'
      then
         begin
             rulo:=1;
             clrscr;
         end
     else rulo:=0;
  end;
end.
```

ANEXA 4.7

Rezultatele rulării programului *param.pas*:

```
fn=     2,5          2,75          3          3,25          3,5    Hz

m= 40.0 kg
k=  15421.26  k=  18659.72  k=  22206.61  k=  26061.92  k=  30225.66
c=    942.48  c=   1036.73  c=   1130.97  c=   1225.22  c=   1319.47
m= 50.0 kg
k=  19276.57  k=  23324.65  k=  27758.26  k=  32577.41  k=  37782.08
c=   1178.10  c=   1295.91  c=   1413.72  c=   1531.53  c=   1649.34
m= 60.0 kg
k=  23131.89  k=  27989.58  k=  33309.91  k=  39092.89  k=  45338.50
c=   1413.72  c=   1555.09  c=   1696.46  c=   1837.83  c=   1979.20
m= 70.0 kg
k=  26987.20  k=  32654.51  k=  38861.57  k=  45608.37  k=  52894.91
c=   1649.34  c=   1814.27  c=   1979.20  c=   2144.14  c=   2309.07
m= 80.0 kg
k=  30842.51  k=  37319.44  k=  44413.22  k=  52123.85  k=  60451.33
c=   1884.96  c=   2073.45  c=   2261.95  c=   2450.44  c=   2638.94
m= 90.0 kg
k=  34697.83  k=  41984.37  k=  49964.87  k=  58639.33  k=  68007.74
c=   2120.58  c=   2332.63  c=   2544.69  c=   2756.75  c=   2968.81
m=100.0 kg
k=  38553.14  k=  46649.30  k=  55516.52  k=  65154.81  k=  75564.16
c=   2356.19  c=   2591.81  c=   2827.43  c=   3063.05  c=   3298.67
```

```
Variatie parametri model functie de raportul c/ccr
----------------------------------------------------
     n[%]       k[N/m]      c[Ns/m]   ccr[Ns/m]      n
----------------------------------------------------
      0.0     31274.31        0.00     3063.05    0.000
      5.0     31352.69      153.34     3066.89    0.050
     10.0     31590.21      307.85     3078.48    0.100
     15.0     31994.18      464.72     3098.10    0.150
     20.0     32577.40      625.24     3126.22    0.200
     25.0     33359.26      790.88     3163.51    0.250
     30.0     34367.36      963.28     3210.95    0.300
     35.0     35640.22     1144.45     3269.87    0.350
     40.0     37231.30     1336.82     3342.06    0.400
     45.0     39215.40     1543.48     3429.96    0.450
     50.0     41699.03     1768.45     3536.91    0.500
     55.0     44837.65     2017.18     3667.60    0.550
     60.0     48866.01     2297.28     3828.81    0.600
     65.0     54154.50     2619.93     4030.68    0.650
     70.0     61321.96     3002.38     4289.12    0.700
     75.0     71483.79     3473.16     4630.89    0.750
     80.0     86872.50     4084.05     5105.07    0.800
     85.0    112699.11     4942.41     5814.61    0.850
     90.0    164598.98     6324.35     7027.07    0.900
     95.0    320750.89     9318.96     9809.45    0.950
    100.0    8230103334.80    1571312.07    1571315.05    1.000
m= 75.0 kg
```

```
 fn=    2,5           2,75          3            3,25          3,5    Hz

 m= 40.0 kg
k=  9869.604  k= 11942.221  k= 14212.230  k= 16679.631  k= 19344.425
 m= 45.0 kg
k= 11103.305  k= 13434.999  k= 15988.759  k= 18764.585  k= 21762.478
 m= 50.0 kg
k= 12337.006  k= 14927.777  k= 17765.288  k= 20849.539  k= 24180.531
 m= 55.0 kg
k= 13570.706  k= 16420.554  k= 19541.817  k= 22934.493  k= 26598.584
 m= 60.0 kg
k= 14804.407  k= 17913.332  k= 21318.346  k= 25019.447  k= 29016.637
 m= 65.0 kg
k= 16038.107  k= 19406.110  k= 23094.874  k= 27104.401  k= 31434.690
 m= 70.0 kg
k= 17271.808  k= 20898.887  k= 24871.403  k= 29189.355  k= 33852.743
 m= 75.0 kg
k= 18505.508  k= 22391.665  k= 26647.932  k= 31274.309  k= 36270.796
 m= 80.0 kg
k= 19739.209  k= 23884.443  k= 28424.461  k= 33359.263  k= 38688.849
 m= 85.0 kg
k= 20972.909  k= 25377.220  k= 30200.989  k= 35444.217  k= 41106.902
 m= 90.0 kg
k= 22206.610  k= 26869.998  k= 31977.518  k= 37529.171  k= 43524.955
 m= 95.0 kg
k= 23440.310  k= 28362.776  k= 33754.047  k= 39614.125  k= 45943.008
 m=100.0 kg
k= 24674.011  k= 29855.553  k= 35530.576  k= 41699.079  k= 48361.062
 m=105.0 kg
k= 25907.712  k= 31348.331  k= 37307.105  k= 43784.033  k= 50779.115
```

ANEXA 5.1

Codul sursă al programului de simulare *kc46.m*, realizat în mediul de programare MathLab este prezentat în continuare:

```
%Program pentru simularea functionarii unui model mecanic
%al corpului omenesc
%stergere ecran in mod text
cla
%stergere ecran in mod grafic
clg
%stergerea variabilelor existente in memorie
clear
disp('   M O D E L    M E C A N I C');
disp('   ~~~~~~~~~    ~~~~~~~~~~~~~');
format compact;
i=sqrt(-1);
%introducerea datelor initiale:
t1=0;
while t1==0
             k1=menu('INITIALIZARI','de   la   TASTATURA','date
implicite','continue');
   if k1==1
     disp('Introduceti datele initiale ale problemei !');
     k12=input('k12=');
     k35=input('k35=');
     k23=input('k23=');
     k34=input('k34=');
     k45=input('k45=');
     k56=input('k56=');
     k57=input('k57=');
     k1f=input('k1f=');
              k21=menu('TIPUL  MODELULUI','cu  amortizare','fara
amortizare');
        if k21==1

     c13=input('c13=');
     c34=input('c34=');
     c45=input('c45=');
     c56=input('c56=');
     c57=input('c57=');

        end;                  %ifk21=1
        if k21==2
           c13=0;c34=0;c45=0;c56=0;c57=0;
                end;                  %ifk21=2

     t6=0;
     while t6==0
```

```
     k6=menu('masa  corpului','toate  masele  componente','doar
masa corpului');
     if k6==1
         m1=input('m1=');
         m2=input('m2=');
         m3=input('m3=');
         m4=input('m4=');
         m5=input('m5=');
         m6=input('m6=');
         m7=input('m7=');
%t1=1;
         t6=1;
     end;                    %ifk6=1
     if k6==2
         m=input('m=');
         m1=6.82*m/100;
         m2=8.52*m/100;
         m3=40.91*m/100;
         m4=1.71*m/100;
         m5=0.57*m/100;
         m6=7.38*m/100;
         m7=34.09*m/100;
         t6=1;
     end;              %ifk6=2
     end;              %while_t6
    end;                    %ifk1=1
    if k1==2;

     m1=5.44;
     m2=6.82;
     m3=32.72;
     m4=1.36;
     m5=0.46;
     m6=5.91;
     m7=27.22;

        k20=menu(' ','cu amortizare','fara amortizare');

  if k20==1
     k12=8527.34;
     k23=8527.34;
     k34=2248.92;
     k45=831.67;
     k56=5654.20;
     k57=34390.09;
     k35=32904.31;
     k1f=6265.10;
     c13=29.39;
     c34=118.33;
     c45=218.53;
     c56=53.38;
     c57=77.88;
      end;                    %ifk20=1

   if k20==2
     k12=8527.34;
     k23=8527.34;
     k34=1690.47;
     k45=1066.71;
     k56=5526.98;
     k57=34046.19;
     k35=33137.39;
     k1f=6228.12;
     c13=0;c34=0;c45=0;c56=0;c57=0;
        end;                   %ifk20=2
    end;                        %ifk1=2

    if k1==3
     t1=1;
    end;                        %ifk1=3
end;                            %while t1
%definirea numaratorilor si numitorilor functiilor de transfer
%corespunzatoare blocurilor componente
n1=1;
d1=1;
n2=k12;
d2=1;
n3=1;
d3=[m1 0];
n4=c13;
d4=1;
n5=c13;
d5=1;
n6=1;
d6=[1 0];
n7=k12;
d7=1;
n8=k12+k23;
d8=1;
n9=1;
d9=[m2 0 0];
n10=k23;
d10=1;
n11=k23;
d11=1;
n12=k35;
d12=1;
n13=k34;
d13=1;
n14=k23+k34+k35;
d14=1;
n15=1;
```

```
d15=[m3 0];
n16=c13+c34;
d16=1;
n17=c34;
d17=1;
n18=1;
d18=[1 0];
n19=k34;
d19=1;
n20=c34;
d20=1;
n21=k45;
d21=1;
n22=k34+k45;
d22=1;
n23=1;
d23=[m4 0];
n24=c34+c45;
d24=1;
n25=c45;
d25=1;
n26=1;
d26=[1 0];
n27=k35;
d27=1;
n28=k45;
d28=1;
n29=c45;
d29=1;
n30=k56;
d30=1;
n31=k35+k45+k56+k57;
d31=1;
n32=1;
d32=[m5 0];
n33=c45+c56+c57;
d33=1;
n34=c56;
d34=1;
n35=c57;
d35=1;
n36=1;
d36=[1 0];
n37=k57;
d37=1;
n38=k56;
d38=1;
n39=k57;
d39=1;
n40=c57;
d40=1;
```

```
n41=1;
d41=[m6 0];
n42=c56;
d42=1;
n43=1;
d43=[1 0];
n44=1;
d44=[m7 0];
n45=1;
d45=[1 0];
n46=k1f;
d46=1;

%definirea numarului de blocuri
nblocks=46;
blkbuild;
%specificarea interconexiunilor dintre blocuri
q=[1 0 0 0 0 0 0 0 0 0
   2 9 0 0 0 0 0 0 0 0
   3 2 -4 5 -7 46 0 0 0 0
   4 3 0 0 0 0 0 0 0 0
   5 15 0 0 0 0 0 0 0 0
   6 3 0 0 0 0 0 0 0 0
   7 6 0 0 0 0 0 0 0 0
   8 9 0 0 0 0 0 0 0 0
   9 7 -8 0 0 0 0 0 0 0
   10 18 0 0 0 0 0 0 0 0
   11 9 0 0 0 0 0 0 0 0
   12 36 0 0 0 0 0 0 0 0
   13 26 0 0 0 0 0 0 0 0
   14 18 0 0 0 0 0 0 0 0
   15 4 11 12 13 -14 -16 17 0 0
   16 15 0 0 0 0 0 0 0 0
   17 23 0 0 0 0 0 0 0 0
   18 15 0 0 0 0 0 0 0 0
   19 18 0 0 0 0 0 0 0 0
   20 15 0 0 0 0 0 0 0 0
   21 36 0 0 0 0 0 0 0 0
   22 26 0 0 0 0 0 0 0 0
   23 19 20 21 -22 -24 25 0 0 0
   24 23 0 0 0 0 0 0 0 0
   25 32 0 0 0 0 0 0 0 0
   26 23 0 0 0 0 0 0 0 0
   27 18 0 0 0 0 0 0 0 0
   28 26 0 0 0 0 0 0 0 0
   29 23 0 0 0 0 0 0 0 0
   30 43 0 0 0 0 0 0 0 0
   31 36 0 0 0 0 0 0 0 0
   32 27 28 29 30 -31 -33 34 35 37
   33 32 0 0 0 0 0 0 0 0
   34 41 0 0 0 0 0 0 0 0
```

```
    35 44 0 0 0 0 0 0 0 0
    36 32 0 0 0 0 0 0 0 0
    37 45 0 0 0 0 0 0 0 0
    38 36 -43 0 0 0 0 0 0 0
    39 36 -45 0 0 0 0 0 0 0
    40 32 -44 0 0 0 0 0 0 0
    41 38 42 0 0 0 0 0 0 0
    42 32 -41 0 0 0 0 0 0 0
    43 41 0 0 0 0 0 0 0 0
    44 39 40 0 0 0 0 0 0 0
    45 44 0 0 0 0 0 0 0 0
    46 1 -6 0 0 0 0 0 0 0];

%specificarea intrarii si iesirilor

iu=[1];
iy=[6 9 18 26 36 43 45];
%interconectarea propriu-zisa
[A B C D]=connect(a,b,c,d,q,iu,iy);
%realizarea unui model minimal
[Am Bm Cm Dm]=minreal(A,B,C,D);
%prin MM-ISI avand matricile Am,Bm,Cm,Dm

iy1=6;
[A1 B1 C1 D1]=connect(a,b,c,d,q,iu,iy1);
[Am1 Bm1 Cm1 Dm1]=minreal(A1,B1,C1,D1);

iy2=9;
[A2 B2 C2 D2]=connect(a,b,c,d,q,iu,iy2);
[Am2 Bm2 Cm2 Dm2]=minreal(A2,B2,C2,D2);

iy3=18;
[A3 B3 C3 D3]=connect(a,b,c,d,q,iu,iy3);
[Am3 Bm3 Cm3 Dm3]=minreal(A3,B3,C3,D3);

iy4=26;
[A4 B4 C4 D4]=connect(a,b,c,d,q,iu,iy4);
[Am4 Bm4 Cm4 Dm4]=minreal(A4,B4,C4,D4);

iy5=36;
[A5 B5 C5 D5]=connect(a,b,c,d,q,iu,iy5);
[Am5 Bm5 Cm5 Dm5]=minreal(A5,B5,C5,D5);

iy6=43;
[A6 B6 C6 D6]=connect(a,b,c,d,q,iu,iy6);
[Am6 Bm6 Cm6 Dm6]=minreal(A6,B6,C6,D6);

iy7=45;
[A7 B7 C7 D7]=connect(a,b,c,d,q,iu,iy7);
[Am7 Bm7 Cm7 Dm7]=minreal(A7,B7,C7,D7);
```

```
[num1 den1]=ss2tf(Am1,Bm1,Cm1,Dm1,iu);
[z1,p1,k]=tf2zp(num1,den1);

[num2 den2]=ss2tf(Am2,Bm2,Cm2,Dm2,iu);
[z2,p2,k]=tf2zp(num2,den2);

[num3 den3]=ss2tf(Am3,Bm3,Cm3,Dm3,iu);
[z3,p3,k]=tf2zp(num3,den3);

[num4 den4]=ss2tf(Am4,Bm4,Cm4,Dm4,iu);
[z4,p4,k]=tf2zp(num4,den4);

[num5 den5]=ss2tf(Am5,Bm5,Cm5,Dm5,iu);
[z5,p5,k]=tf2zp(num5,den5);

[num6 den6]=ss2tf(Am6,Bm6,Cm6,Dm6,iu);
[z6,p6,k]=tf2zp(num6,den6);

[num7 den7]=ss2tf(Am7,Bm7,Cm7,Dm7,iu);
[z7,p7,k]=tf2zp(num7,den7);

Am1;
%pause
Bm1;
%pause
Cm1;
%pause
Dm1;
%pause

w=logspace(-1,3,100);

t11=0;
while t11==0

k11=menu('CARACTERISTICI_FRECVENTA','H1','H2','H3','H4','H5','H
6','H7','PE RAND','SUPRAPUSE','CONTINUE');
%Hi=functii de transfer=zi/u
    if k11==1

    if (real(p1)<=0.00001),disp('Sistem stabil')
           else,  disp(''),disp('      Pentru  H1(s)  exista  poli
nenegativi')
         disp('                     ==>  sistemul  este  instabil
!!!');
    end            %if
    [re1,im1]=nyquist(num1,den1,w);
    [mod1,arg1]=bode(num1,den1,w);
    modd1=20*log10(mod1);
```

```
semilogx(w,modd1,'w'),grid,title('c.a.p.1'),xlabel('pulsatia'),
ylabel('Ampl 1');
    pause
    if (real(p1)<0),loglog(w,modd1),grid;
       title('c.a.p.1'),xlabel('pulsatia'),ylabel('Ampl 1');
    end;               %if

semilogx(w,arg1),grid,title('c.f.p.1'),xlabel('pulsatia'),ylabe
l('Arg 1');
    pause

end;                    %ifk11=1
if k11==2


if (real(p2)<=0.00001),disp('Sistem stabil')
         else,  disp(''),disp('       Pentru  H2(s)  exista  poli
nenegativi')
     disp('                   ==> sistemul este instabil !!!');
end             %if
[re2,im2]=nyquist(num2,den2,w);
[mod2,arg2]=bode(num2,den2,w);
modd2=20*log10(mod2);
semilogx(w,modd2,'w'),grid,title('c.a.p.2'),xlabel('pulsatia'),
ylabel('Ampl 2');
pause
if (real(p2)<0),loglog(w,modd2),grid;
  title('c.a.p.2'),xlabel('pulsatia'),ylabel('Ampl 2');
end;               %if
semilogx(w,arg2),grid,title('c.f.p.2'),xlabel('pulsatia'),ylabe
l('Arg 2');
pause

end                  %ifk11=2
if k11==3

if (real(p3)<=0.00001),disp('Sistem stabil')
         else,  disp(''),disp('       Pentru  H3(s)  exista  poli
nenegativi')
     disp('                   ==> sistemul este instabil !!!');
end             %if
[re3,im3]=nyquist(num3,den3,w);
[mod3,arg3]=bode(num3,den3,w);
modd3=20*log10(mod3);
semilogx(w,modd3,'w'),grid,title('c.a.p.3'),xlabel('pulsatia'),
ylabel('Ampl 3');
pause
if (real(p3)<0),loglog(w,modd3),grid;
  title('c.a.p.3'),xlabel('pulsatia'),ylabel('Ampl 3');
end;               %if
```

```
semilogx(w,arg3),grid,title('c.f.p.3'),xlabel('pulsatia'),ylabe
l('Arg 3');
pause

end                  %ifk11=3
if k11==4

if (real(p4)<=0.00001),disp('Sistem stabil')
         else,  disp(''),disp('       Pentru  H4(s)  exista  poli
nenegativi')
     disp('                   ==> sistemul este instabil !!!');
end             %if
[re4,im4]=nyquist(num4,den4,w);
[mod4,arg4]=bode(num4,den4,w);
modd4=20*log10(mod4);
semilogx(w,modd4,'w'),grid,title('c.a.p.4'),xlabel('pulsatia'),
ylabel('Ampl 4');
pause
if (real(p4)<0),loglog(w,modd4),grid;
  title('c.a.p.4'),xlabel('pulsatia'),ylabel('Ampl 4');
end;               %if
semilogx(w,arg4),grid,title('c.f.p.4'),xlabel('pulsatia'),ylabe
l('Arg 4');
pause

end                  %ifk11=4
if k11==5

if (real(p5)<=0.00001),disp('Sistem stabil')
         else,  disp(''),disp('       Pentru  H5(s)  exista  poli
nenegativi')
     disp('                   ==> sistemul este instabil !!!');
end             %if
[re5,im5]=nyquist(num5,den5,w);
[mod5,arg5]=bode(num5,den5,w);
modd5=20*log10(mod5);
semilogx(w,modd5,'w'),grid,title('c.a.p.5'),xlabel('pulsatia'),
ylabel('Ampl 5');
pause
if (real(p5)<0),loglog(w,modd5),grid;
  title('c.a.p.5'),xlabel('pulsatia'),ylabel('Ampl 5');
end;               %if
semilogx(w,arg5),grid,title('c.f.p.5'),xlabel('pulsatia'),ylabe
l('Arg 5');
pause

end                  %ifk11=5
if k11==6


if (real(p6)<=0.00001),disp('Sistem stabil')
```

```
         else,  disp(''),disp('       Pentru  H6(s)  exista  poli
nenegativi')
     disp('                   ==> sistemul este instabil !!!');
end            %if
[re6,im6]=nyquist(num6,den6,w);
[mod6,arg6]=bode(num6,den6,w);
modd6=20*log10(mod6);
semilogx(w,modd6,'w'),grid,title('c.a.p.6'),xlabel('pulsatia'),
ylabel('Ampl 6');
pause
if (real(p6)<0),loglog(w,modd6),grid;
  title('c.a.p.6'),xlabel('pulsatia'),ylabel('Ampl 6');
end;              %if
semilogx(w,arg6),grid,title('c.f.p.6'),xlabel('pulsatia'),ylabe
l('Arg 6');
pause

end                  %ifk11=6
if k11==7

if (real(p7)<=0.00001),disp('Sistem stabil')
         else,  disp(''),disp('       Pentru  H7(s)  exista  poli
nenegativi')
     disp('                   ==> sistemul este instabil !!!');
end            %if
[re7,im7]=nyquist(num7,den7,w);
[mod7,arg7]=bode(num7,den7,w);
modd7=20*log10(mod7);
semilogx(w,modd7,'w'),grid,title('c.a.p.7'),xlabel('pulsatia'),
ylabel('Ampl 7');
pause
if (real(p7)<0),loglog(w,modd7),grid;
  title('c.a.p.7'),xlabel('pulsatia'),ylabel('Ampl 7');
end;              %if
semilogx(w,arg7),grid,title('c.f.p.7'),xlabel('pulsatia'),ylabe
l('Arg 7');
pause

end                 %ifk11=7

if k11==8
     hold off
w=logspace(1,2.5,100);

t100=0;
while t100==0
                        k100=menu('Caracteristici_frecventa','Pe
rand','H1','H2','H3','H4','H5','H6','H7','Suprapuse','Revenire'
);
%Hi=functii de transfer=zi/u
  if k100==1
```

```
    hold off
    [mod1,arg1]=bode(num1,den1,w);
semilogx(w,mod1,'r'),grid,title('c.a.p.1'),xlabel('pulsatia'),
ylabel('Ampl');pause
    hold on
    [mod2,arg2]=bode(num2,den2,w);
semilogx(w,mod2,'b'),grid,title('c.a.p.1,2'),xlabel('pulsatia')
,ylabel('Ampl');pause
    [mod3,arg3]=bode(num3,den3,w);
semilogx(w,mod3,'y'),grid,title('c.a.p.1,2,3'),xlabel('pulsatia'),
ylabel('Ampl');pause
    [mod4,arg4]=bode(num4,den4,w);
semilogx(w,mod4,'m'),grid,title('c.a.p.1,2,3,4'),xlabel('pulsatia
'),ylabel('Ampl');pause
    [mod5,arg5]=bode(num5,den5,w);
semilogx(w,mod5,'c'),grid,title('c.a.p.1,2,3,4,5'),xlabel('pulsa
tia'),ylabel('Ampl');pause
    [mod6,arg6]=bode(num6,den6,w);
semilogx(w,mod6,'w'),grid,title('c.a.p.1,2,3,4,5,6'),xlabel('pulsa
tia'),ylabel('Ampl');pause
    [mod7,arg7]=bode(num7,den7,w);
semilogx(w,mod7,'r'),grid,title('c.a.p.1,2,3,4,5,6,7'),xlabel
('pulsatia'),ylabel('Ampl');pause
    hold off

       end     %ifk100=1

       if k100==2
         [mod1,arg1]=bode(num1,den1,w);

     semilogx(w,mod1,'r'),grid,title('c.a.p.1'),xlabel('pulsatia'),y
     label('Ampl');pause
       end     %ifk100=2
       if k100==3
         [mod2,arg2]=bode(num2,den2,w);

     semilogx(w,mod2,'b'),grid,title('c.a.p.2'),xlabel('pulsatia'),y
     label('Ampl');pause
       end     %ifk100=3
       if k100==4

         [mod3,arg3]=bode(num3,den3,w);

     semilogx(w,mod3,'y'),grid,title('c.a.p.3'),xlabel('pulsatia'),y
     label('Ampl');pause
       end      %ifk100=4
       if k100==5

         [mod4,arg4]=bode(num4,den4,w);
```

```
semilogx(w,mod4,'m'),grid,title('c.a.p.4'),xlabel('pulsatia'),y
label('Ampl');pause
  end      %ifk100=5
  if k100==6

    [mod5,arg5]=bode(num5,den5,w);

semilogx(w,mod5,'c'),grid,title('c.a.p.5'),xlabel('pulsatia'),y
label('Ampl');pause
  end      %ifk100=6
  if k100==7

    [mod6,arg6]=bode(num6,den6,w);

semilogx(w,mod6,'w'),grid,title('c.a.p.6'),xlabel('pulsatia'),y
label('Ampl');pause
  end      %ifk100=7
  if k100==8

    [mod7,arg7]=bode(num7,den7,w);

semilogx(w,mod7,'r'),grid,title('c.a.p.7'),xlabel('pulsatia'),y
label('Ampl');pause

  end      %ifk100=8
  if k100==9

    hold off
    [mod1,arg1]=bode(num1,den1,w);

semilogx(w,mod1,'r'),grid,title('c.a.p.1'),xlabel('pulsatia'),y
label('Ampl');
    hold on
    [mod2,arg2]=bode(num2,den2,w);

semilogx(w,mod2,'b'),grid,title('c.a.p.1,2'),xlabel('pulsatia')
,ylabel('Ampl');

    [mod3,arg3]=bode(num3,den3,w);

semilogx(w,mod3,'y'),grid,title('c.a.p.1,2,3'),xlabel('pulsatia
'),ylabel('Ampl');

    [mod4,arg4]=bode(num4,den4,w);

semilogx(w,mod4,'m'),grid,title('c.a.p.1,2,3,4'),xlabel('pulsat
ia'),ylabel('Ampl');

    [mod5,arg5]=bode(num5,den5,w);
```

```
semilogx(w,mod5,'c'),grid,title('c.a.p.1,2,3,4,5'),xlabel('puls
atia'),ylabel('Ampl');

    [mod6,arg6]=bode(num6,den6,w);

semilogx(w,mod6,'w'),grid,title('c.a.p.1,2,3,4,5,6'),xlabel('pu
lsatia'),ylabel('Ampl');

    [mod7,arg7]=bode(num7,den7,w);

semilogx(w,mod7,'r'),grid,title('c.a.p.1,2,3,4,5,6,7'),xlabel('
pulsatia'),ylabel('Ampl');pause
    hold off
  end     %ifk100=9

  if k100==10
     t100=1;
  end     %ifk100=10
end     %while_t100

end                 %ifk11=8


if k11==9
     hold off
w=logspace(1,2.5,100);
     [mod1,arg1]=bode(num1,den1,w);
modd1=20*log10(mod1);

semilogx(w,modd1,'w'),grid,title('c.a.p.1'),xlabel('pulsatia'),
ylabel('Ampl');

    hold on
    [mod2,arg2]=bode(num2,den2,w);
modd2=20*log10(mod2);

semilogx(w,modd2,'w'),grid,title('c.a.p.2'),xlabel('pulsatia'),
ylabel('Ampl');

    [mod3,arg3]=bode(num3,den3,w);
modd3=20*log10(mod3);

semilogx(w,modd3,'w'),grid,title('c.a.p.3'),xlabel('pulsatia'),
ylabel('Ampl');

    [mod4,arg4]=bode(num4,den4,w);
modd4=20*log10(mod4);

semilogx(w,modd4,'w'),grid,title('c.a.p.4'),xlabel('pulsatia'),
ylabel('Ampl');
```

```
    [mod5,arg5]=bode(num5,den5,w);
modd5=20*log10(mod5);

semilogx(w,modd5,'w'),grid,title('c.a.p.5'),xlabel('pulsatia'),
ylabel('Ampl');

    [mod6,arg6]=bode(num6,den6,w);
modd6=20*log10(mod6);

semilogx(w,modd6,'w'),grid,title('c.a.p.6'),xlabel('pulsatia'),
ylabel('Ampl');

    [mod7,arg7]=bode(num7,den7,w);
modd7=20*log10(mod7);

semilogx(w,modd7,'w'),grid,title('c.a.p.7'),xlabel('pulsatia'),
ylabel('Ampl');pause

    hold off

end                    %ifk11=9

if k11==10;
  t11=1;
end;                   %ifk11=10
end;                   %while_t11

%Generare semnal de intrare: semnal sinusoidal
%n=input('Dati lungimea secventei  n=');
%F0=input('Dati amplitudinea fortei F0=');

tfin=0;
while tfin==0

t2=0;
while t2==0
                                                 k2=menu('SEMNAL
INTRARE','SINUSOIDAL','IMPULS','TREAPTA','CONTINUE');
    if k2==1
     k6=menu('init','de la tastatura','implicite');
     if k6==1
         n=input('Dati lungimea secventei  n=');
         x0=input('Dati amplitudinea miscarii x0=');
         F0=k1f*x0;
         k7=menu('Dati','pulsatia','frecventa');
           if k7==1
          omega=input('Dati pulsatia:');
          niu=omega/(2*pi);         %frecventa
           end;                          %ifk7=1
           if k7==2
          niu=input('Dati frecventa, in Hz:  ');

           end;                              %ifk7=2

     end;                %ifk6=1
     if k6==2
         n=3;
         x0=0.01;
         F0=k1f*0.01;
         niu=10;
     end;                %ifk6=2
     T=1/niu;            %calculul perioadei oscilatiilor
     i=0:0.01:n;
     u=x0*sin(2*pi*i./T);
     p=u';
        %u=u/k8;
     plot(0:0.01:n,u),grid,title('Semnal intrare  xf - sinus');
     xlabel('t'),ylabel('x0');
     pause;

[y,t]=lsim(Am,Bm,Cm,Dm,p,0:0.01:n);

[y1,t]=lsim(Am1,Bm1,Cm1,Dm1,p,0:0.01:n);
[y2,t]=lsim(Am2,Bm2,Cm2,Dm2,p,0:0.01:n);
[y3,t]=lsim(Am3,Bm3,Cm3,Dm3,p,0:0.01:n);
[y4,t]=lsim(Am4,Bm4,Cm4,Dm4,p,0:0.01:n);
[y5,t]=lsim(Am5,Bm5,Cm5,Dm5,p,0:0.01:n);
[y6,t]=lsim(Am6,Bm6,Cm6,Dm6,p,0:0.01:n);
[y7,t]=lsim(Am7,Bm7,Cm7,Dm7,p,0:0.01:n);

    end;                    %ifk2=1

    if k2==2
     k6=menu('VALORI INPUT','de la tastatura','implicite');
     if k6==1
         n=input('Dati lungimea secventei  n=');
         F0=input('Dati amplitudinea fortei F0=');
         b=input('Dati lungimea impulsului b<n,   b=');
     end;                %ifk6=1
     if k6==2
         n=10;
         F0=200;
         b=1;
     end;                %ifk6=2
     u=zeros(1,n);
        for i=1:0.01:b
          u(i)=F0;
          p=u';
          end;                %for
     plot(1:n,u),grid,title('FORTA PERTURBATOARE  - impuls');
     xlabel('t'),ylabel('F');
     pause;

[y,t]=lsim(Am,Bm,Cm,Dm,p,1:n);

[y1,t]=lsim(Am1,Bm1,Cm1,Dm1,p,1:n);
[y2,t]=lsim(Am2,Bm2,Cm2,Dm2,p,1:n);
[y3,t]=lsim(Am3,Bm3,Cm3,Dm3,p,1:n);
[y4,t]=lsim(Am4,Bm4,Cm4,Dm4,p,1:n);
[y5,t]=lsim(Am5,Bm5,Cm5,Dm5,p,1:n);
[y6,t]=lsim(Am6,Bm6,Cm6,Dm6,p,1:n);
[y7,t]=lsim(Am7,Bm7,Cm7,Dm7,p,1:n);

    end;                    %ifk2=2

    if k2==3
     k6=menu('init','de la tastatura','implicite');
     if k6==1
         n=input('Dati lungimea secventei  n=');
         F0=input('Dati amplitudinea fortei F0=');
     end;                 %ifk6=1
     if k6==2
         n=10;
         F0=1;
     end;                 %ifk6=2
for i=0:n
          u(i)=F0;
          p=u';
          end;                 %for

     plot(0:n,u),grid,title('FORTA PERTURBATOARE - treapta');
     xlabel('t'),ylabel('F');
     pause;

[y,t]=lsim(Am,Bm,Cm,Dm,p,0:n);

[y1,t]=lsim(Am1,Bm1,Cm1,Dm1,p,0:n);
[y2,t]=lsim(Am2,Bm2,Cm2,Dm2,p,0:n);
[y3,t]=lsim(Am3,Bm3,Cm3,Dm3,p,0:n);
[y4,t]=lsim(Am4,Bm4,Cm4,Dm4,p,0:n);
[y5,t]=lsim(Am5,Bm5,Cm5,Dm5,p,0:n);
[y6,t]=lsim(Am6,Bm6,Cm6,Dm6,p,0:n);
[y7,t]=lsim(Am7,Bm7,Cm7,Dm7,p,0:n);

    end;                    %ifk2=3

    if k2==4
     t2=1;
    end;                    %ifk2=4
end;                        %while_t2
```

```
%afisarea grafica a raspunsului
t3=0;
while t3==0
k3=menu('AFISARE','un parametru','all','all1', 'all2','all3',
'all4', 'REVENIRE','EXIT');
    if k3==1
     t4=0;
     while t4==0

k4=menu('PARAMETRI','z1','z2','z3','z4','z5','z6','z7','Quit');
         if k4==1
          plot(y1);grid;xlabel('t');ylabel('z1');
          pause;
         end;            %ifk4=1
         if k4==2
          plot(y2),grid,xlabel('t'),ylabel('z2');
          pause;
         end;            %ifk4=2
         if k4==3
          plot(y3),grid,xlabel('t'),ylabel('z3');
          pause;
         end;            %ifk4=3
         if k4==4
          plot(y4);grid;xlabel('t');ylabel('z4');
          pause;
         end;            %ifk4=4
         if k4==5
          plot(y5);grid;xlabel('t');ylabel('z5');
          pause;
         end;            %ifk4=5
         if k4==6
          plot(y6);grid;xlabel('t');ylabel('z6');
          pause;
         end;            %ifk4=6
         if k4==7
          plot(y7);grid;xlabel('t');ylabel('z7');
          pause;
         end;            %ifk4=7
         if k4==8
          t4=1;
         end;            %ifk4=8
     end;                %while_t4
    end;                    %ifk3=1
    if k3==2
        %hold on
     plot(y);grid;xlabel('t');ylabel('y');
     pause;
    end;                    %ifk3=2
    if k3==3
     subplot(111),plot(y1,'w'),grid,xlabel('t'),ylabel('z1'),pa
use;
     plot(y2,'w'),grid,xlabel('t'),ylabel('z2');pause;
plot(y3,'w'),grid,xlabel('t'),ylabel('z3');pause;
plot(y4,'w'),grid,xlabel('t'),ylabel('z4');pause;
plot(y5,'w'),grid,xlabel('t'),ylabel('z5');pause;
plot(y6,'w'),grid,xlabel('t'),ylabel('z6');pause;
plot(y7,'w'),grid,xlabel('t'),ylabel('z7');pause;
end;                    %ifk3=3

    if k3==4

subplot(221),plot(y1,'w'),grid;xlabel('t');ylabel('z1');pause
subplot(222),plot(y2,'w'),grid;xlabel('t');ylabel('z2');pause
subplot(223),plot(y3,'w'),grid;xlabel('t');ylabel('z3');pause
subplot(224),plot(y4,'w'),grid;xlabel('t');ylabel('z4');pause
       clg;
subplot(221),plot(y5,'w'),grid;xlabel('t');ylabel('z5');pause
subplot(222),plot(y6,'w'),grid;xlabel('t');ylabel('z6');pause
subplot(223),plot(y7,'w'),grid;xlabel('t');ylabel('z7');
     pause;
     subplot(111);
     clg

    end;                    %ifk3=4

    if k3==5
        subplot(1,1,1);
        hold off;
        plot(u,'w');
     hold on
     plot(y);grid;xlabel('t');ylabel('y');
     pause;
        hold off;
    end;                    %ifk3=5

    if k3==6
        hold off
        hold on;
     subplot(111)
     plot(u,'w');
     pause
        hold on
     plot(y1,'m'),grid,xlabel('t'),ylabel('..+z1'),pause;
     pause
     plot(y2,'r'),grid,xlabel('t'),ylabel('..+z2');pause;
     plot(y3,'r'),grid,xlabel('t'),ylabel('..+z3');pause;
     plot(y4,'c'),grid,xlabel('t'),ylabel('..+z4');pause;
     plot(y5,'y'),grid,xlabel('t'),ylabel('..+z5');pause;
     plot(y6,'b'),grid,xlabel('t'),ylabel('..+z6');pause;
     plot(y7,'w'),grid,xlabel('t'),ylabel('..+z7');pause;
     hold off
 end;                    %ifk3=6

    if k3==7
     t3=1;

     end   %ifk3=7

    if k3==8
     t3=1;
     tfin=1;

    end;                        %ifk3=8
end;                            %while_t3
end;   %while_tfin

cla;
disp(''),disp(''),disp(''),disp(''),disp(''),disp(''),disp(''),
disp('');
disp(''),disp('');
disp('                             VA  MULTUMIM  CA  ATI  FOLOSIT  ACEST
PROGRAM');
disp(''),disp(''),disp(''),disp(''),disp(''),disp('');
disp('                                                    prof.coord.:
doctorand:');
disp('                         prof.univ.dr.ing.Liviu          BRINDEU
ing.Cornel POPA');
disp(''),disp(''),disp(''),disp(''),disp(''),disp('');
end                            %program
```

ANEXA 5.2

Modificările operate asupra programului MathLab numit *kc46.m* pentru a deveni *mm23.m*:

```
%diferenta dintre mm45.m si mm23.m:
n1=1;
d1=1;
n2=1;
d2=[m1 c13 k12];
n3=k12;
d3=[m2 0 k12+k23];
n4=k23;
d4=[m3 c13+c34 k23+k34+k35];
n5=[c34 k34];
d5=[m4 c34+c45 k34+k45];
n6=[c45 k45];
d6=[m5 c45+c56+c57 k35+k45+k56+k57];
n7=[c56 k56];
d7=[m6 c56 k56];
n8=[c57 k57];
d8=[m7 c57 k57];
n9=k12;
d9=1;
n10=[c13 0];
d10=[m1 c13 k12];
n11=k23;
d11=[m2 0 k12+k23];
n12=[c13 0];
d12=[m3 c13+c34 k23+k34+k35];
n13=[c34 k34];
d13=[m3 c13+c34 k23+k34+k35];
n14=k35;
d14=[m3 c13+c34 k23+k34+k35];
n15=k35;
d15=[m5 c45+c56+c57 k35+k45+k56+k57];
n16=[c45 k45];
d16=[m4 c34+c45 k34+k45];
n17=[c56 k56];
d17=[m5 c45+c56+c57 k35+k45+k56+k57];
n18=[c57 k57];
d18=[m5 c45+c56+c57 k35+k45+k56+k57];
n19=1;
d19=1;
n20=1;
d20=1;
n21=1;
d21=1;
n22=1;
```

```
d22=1;
n23=1;
d23=1;
%definirea numarului de blocuri
nblocks=23;
blkbuild;
%specificarea interconexiunilor dintre blocuri
q=[1 0 0 0 0
   2 1 9 0 0
   3 19 0 0 0
   4 20 0 0 0
   5 21 0 0 0
   6 22 0 0 0
   7 23 0 0 0
   8 23 0 0 0
   9 20 0 0 0
   10 21 0 0 0
   11 21 0 0 0
   12 19 0 0 0
   13 22 0 0 0
   14 23 0 0 0
   15 21 0 0 0
   16 23 0 0 0
   17 7 0 0 0
   18 8 0 0 0
   19 2 10 0 0
   20 3 11 0 0
   21 4 12 13 14
   22 5 16 0 0
   23 6 15 17 18];
%specificarea intrarii si iesirilor
iu=[1];
iy=[19 20 21 22 23 7 8];

%interconectarea propriu-zisa
[A B C D]=connect(a,b,c,d,q,iu,iy);
%realizarea unui model minimal
[Am Bm Cm Dm]=minreal(A,B,C,D);
%prin MM-ISI avand matricile Am,Bm,Cm,Dm

iy1=19;
[A1 B1 C1 D1]=connect(a,b,c,d,q,iu,iy1);
[Am1 Bm1 Cm1 Dm1]=minreal(A1,B1,C1,D1);
iy2=20;
[A2 B2 C2 D2]=connect(a,b,c,d,q,iu,iy2);
[Am2 Bm2 Cm2 Dm2]=minreal(A2,B2,C2,D2);
iy3=21;
[A3 B3 C3 D3]=connect(a,b,c,d,q,iu,iy3);
[Am3 Bm3 Cm3 Dm3]=minreal(A3,B3,C3,D3);
iy4=22;
[A4 B4 C4 D4]=connect(a,b,c,d,q,iu,iy4);
```

```
[Am4 Bm4 Cm4 Dm4]=minreal(A4,B4,C4,D4);
iy5=23;
[A5 B5 C5 D5]=connect(a,b,c,d,q,iu,iy5);
[Am5 Bm5 Cm5 Dm5]=minreal(A5,B5,C5,D5);
iy6=7;
[A6 B6 C6 D6]=connect(a,b,c,d,q,iu,iy6);
[Am6 Bm6 Cm6 Dm6]=minreal(A6,B6,C6,D6);
iy7=8;
[A7 B7 C7 D7]=connect(a,b,c,d,q,iu,iy7);
[Am7 Bm7 Cm7 Dm7}=minreal(A7,B7,C7,D7);
```

ANEXA 5.3

Codul sursă al programului de simulare *sim.m*, realizat în mediul de programare MathLab este prezentat în continuare:

```
     %Program pentru simularea functionarii modelului mecanic
simplificat
     %al corpului omenesc
     %stergere ecran in mod text
     cla
     %stergere ecran in mod grafic
     clg
     %stergerea variabilelor existente in memorie
     clear
     disp('   M O D E L    M E C A N I C');
     disp('   ~~~~~~~~~    ~~~~~~~~~~~~~');
     format compact;
     i=sqrt(-1);
     %introducerea datelor initiale:

     t0=0;
     while t0==0

     t1=0;
     while t1==0
             k1=menu('INIT','de la TASTATURA', 'date implicite',
'continue');
         if k1==1
          disp('Introduceti datele initiale ale problemei !');
             k21=menu(' ','cu amortizare','fara amortizare');
             if k21==1
              k=input('k=');
                 c=input('c=');
                 m=input('m=');
             end;                  %ifk21=1
             if k21==2
              k=input('k=');
                 c=0;
                 m=input('m=');
             end;                  %ifk21=2
         end;                  %ifk1=1
         if k1==2;
             m=65;
             k20=menu(' ','cu amortizare','fara amortizare');
                 if k20==1
                    k=23328;
                    c=246;
                 end;                  %ifk20=1
                 if k20==2
                        k=23095;
```

```
                         c=0;
                 end;               %ifk20=2
         end;                   %ifk1=2
         if k1==3
          t1=1;
         end;                   %ifk1=3
     end;                       %while t1
     %se introduce sistemul prin expresiile polinoamelor de la
numaratorul
     %    si numitorul functiei de transfer
     num=1;
     den=[m c k];
     %definirea numarului de blocuri

     %determinarea matricilor MM-ISI
     [A B C D]=tf2ss(num,den);
     %determinarea configuratiei poli-zerouri
     [z,p,k]=tf2zp(num,den);

     t4=0;
     while t4==0
                k4=menu('Model matematic','MM-ISI','MM-II','poli-
zerouri','CONTINUE');
         if k4==1
            disp('');disp('');
            disp('Matricele MM-ISI sunt:');disp('');
            A,disp('');
            B,disp('');
            C,disp('');
            D,disp('');
            pause
         end                  %ifk4=1
         if k4==2
            disp('');disp('');
             disp('Coeficientii numaratorului si numitorului f.d.t.
sunt:');
            disp('');
            num,disp('');
            den,disp('');
            pause
         end                  %ifk4=2
         if k4==3
            disp('');disp('');
            disp('Polii si zerourile f.d.t. sunt:');disp('');
            p,disp('');
            z
            pause
         end                  %ifk4=3
         if k4==4
            t4=1;
         end                  %ifk4=4
     end                      %while_t4=0

     w=logspace(-1,3,50);
     [re,im]=nyquist(num,den,w);
     %plot(re,im),grid,title('Hodograf        Nyquist        pentru
H(s)');pause;
     [mod,arg]=bode(num,den,w);
     modd=20*log10(mod);
     if  (real(p)<=0),disp(''),disp('                Sistemul  este
stabil.')
            else,  disp(''),disp('       Pentru  H(s)  exista  poli
nenegativi')
              disp('                     ==> sistemul este instabil
!!!');
     end                %if
     pause;

     t11=0;
     while t11==0

k11=menu('CARACTERISTICI_FRECVENTA','c.a.p.','c.f.p.','CONTINUE');
         if k11==1
                 %afisarea grafica a caracteristicii logaritmice
amplitudine pulsatie
          semilogx(w,modd),grid,title('c.a.p.'),xlabel('pulsatia')
,ylabel('Ampl');
          if (real(p)<0),pause,loglog(w,modd),grid;
               title('c.a.p.'),xlabel('pulsatia'),ylabel('Ampl');
             end;              %if
         end                   %k11=1
         if k11==2
              %afisarea grafica a caracteristicii logaritmice faza
pulsatie
             semilogx(w,arg), grid, title('c.f.p.'),
xlabel('pulsatia'), ylabel('Arg');
          pause
         end                   %k11=2
         if k11==3;
             t11=1;
         end;                      %ifk11=3
     end;                          %while_t11

     %Generare  semnal  de  intrare:  semnal  sinusoidal,  treapta  si
impuls
     % si obtinerea raspunsului sistemului; simulare
     t2=0;
     while t2==0
                                                    k2=menu('SEMNAL
INTRARE','SINUSOIDAL','IMPULS','TREAPTA','CONTINUE');
         if k2==1
          k6=menu('init','de la tastatura','implicite');
          if k6==1
              n=input('Dati lungimea secventei  n=');
               x0=input('Dati amplitudinea miscarii perturbatoare
x0=');
              k7=menu('Dati','pulsatia','frecventa');
                if k7==1
                     %t100=0;
                     omega=input('Dati pulsatia:');
               %while t100==0
                end;                          %ifk7=1
                if k7==2
               niu=input('Dati frecventa, in Hz:  ');
     end;                            %ifk7=2
          end;                  %ifk6=1
          if k6==2
              n=10;
              x0=.01;
              niu=2.5;
          end;                  %ifk6=2

          T=1/niu;              %calculul perioadei oscilatiilor
          i=1:0.01:n;
          %u=F0*sin(2*pi*i./T);
          u1=x0*sin(2*pi*i./T);
          %u=k*x0*sin(2*pi*i./T);
          u=k*x0*sin(2*pi*i./T)+c*x0*(2*pi*niu)*cos(2*pi*i./T);
          %u=k*x0*cos(2*pi*i./T)-c*x0*(2*pi*niu)*sin(2*pi*i./T);
          k101=menu('Afisarea variatiei miscarii perturbatoare','D
A','N U');
                 if k101==1
                    plot(1:0.01:n,u1,'r'),grid,title('MISCAREA
PERTURBATOARE');
               xlabel('t'),ylabel('xf'),pause
                 end                    %ifk101=1
              if k101==2
               t201=1
              end                    %ifk101=2
          for i=1:n
              t(i)=i;
            end;                    %for
             p=u';
          [y,t]=lsim(A,B,C,D,p,1:0.01:n);
         end;                       %ifk2=1

         if k2==2
          k6=menu('init','de la tastatura','implicite');
          if k6==1
              n=input('Dati lungimea secventei  n=');
              F0=input('Dati amplitudinea fortei F0=');
              b=input('Dati lungimea impulsului b<n,   b=');
          end;                  %ifk6=1
          if k6==2
              n=20;
              F0=1;
              b=1;
          end;                %ifk6=2
          u=zeros(1,n);
          for i=1:0.01:b
               u(i)=F0;
                    end;                %for
          plot(1:n,u,'r'),grid,title('FORTA PERTURBATOARE');
          xlabel('t'),ylabel('F');
          pause;
          [y,t]=lsim(A,B,C,D,u,1:n);
         end;                   %ifk2=2
         if k2==3
          k6=menu('init','de la tastatura','implicite');
          if k6==1
              n=input('Dati lungimea secventei  n=');
              F0=input('Dati amplitudinea fortei F0=');
          end;                %ifk6=1
          if k6==2
              n=20;
              F0=1;
          end;                %ifk6=2
             for i=1:n
               u(i)=F0;
             end;               %for
          plot(1:n,u,'r'),grid,title('FORTA PERTURBATOARE');
          xlabel('t'),ylabel('F');
          pause;
          [y,t]=lsim(A,B,C,D,u,1:n);
         end;                   %ifk2=3
         if k2==4
          t2=1;
         end;                   %ifk2=4
     end;                       %while_t2
     %t=1:n;
     %afisarea grafica a raspunsului
     t3=0;

     while t3==0
         k3=menu('AFISARE','parametru','REVENIRE','EXIT');
         if k3==1
          comet(y);
          plot(y,'r');grid;xlabel('t');ylabel('y');
%uuu
         end;              %ifk3=1                      %uuu

         if k3==2
          t3=1;
         end;                   %ifk3=2
         if k3==3
          t3=1;
          t0=1;
         end;                       %ifk3=3
     end;                           %while_t3

     end;                     %while_t0

     cla;
     disp(''),disp(''),disp(''),disp(''),disp(''),disp(''),disp(''
),disp('');
     disp(''),disp('');
     disp('                     VA MULTUMIM CA ATI FOLOSIT ACEST
PROGRAM');
     disp(''),disp(''),disp(''),disp(''),disp(''),disp('');
     disp('                                                 prof.coord.:
doctorand:');
     disp('                  prof.univ.dr.ing.Liviu          BRANDEU
dipl.ing.Cornel POPA');
     disp(''),disp(''),disp(''),disp(''),disp(''),disp('');
     end                            %program
```

ANEXA 6.1

Codul sursă al programului Pascal, *elastsim.pas*, este:

```
     program estimare_constante_elastice;
     uses crt,printer;
     var men1,men2:integer;
         m,s1,m1,m2,m3,m4,m5,m6,m7,m8,m9,m10,r:real;
         pr1,pr2,pr3,pr4,pr5,pr6,pr7,pr8,pr9,pr10,pr:real;
         fn1,fn2,fn3,fn4,fn5,fn6,fn7,fn8,fn9,fn10,a,b,c:real;
         omega1,omega2,omega3,omega4,omega5,omega6,omega7,omega8,
omega9,omega10:real;
         k12,k23,k34,k35,k45,k56,k57,k,k58,k39,k910:real;
         ch,vm,vp,vf,vimp,cp,st,im:char;

     procedure tasta1;
          begin
               ch:=readkey;
          end;

     procedure antet;
          begin
               textbackground(3);
               clrscr;
               textbackground(1);
               textcolor(3);
               gotoxy(20,10);
               writeln('   M   O   D   E   L     M   E   C   A   N   I
C');
               gotoxy(10,12);
               writeln('al  corpului  omenesc  asezat  Œn  picioare  pe
o platformt');
               writeln;
               delay(2050);
               clrscr;
               textbackground(3);
               textcolor(1);
               gotoxy(12,3);
               writeln('  D E T E R M I N A R E   C O N S T A N T
E   E L A S T I C E');
               writeln;
               writeln;
               tasta1;
          end;

     procedure tasta;
          begin
               gotoxy(23,20);
               writeln('>> APASATI ORICE TASTA PENTRU A COMTINUA!
<<');
```

```
               ch:=readkey;
          end;

     procedure meniu1;
          begin
               writeln('       D A T E   I N I T I A L E');
               writeln('       ~~~~~~~~~~~~~~~~~~~~~~~~~');
               writeln('  Tasteaz†:');
               writeln;
               writeln('        1 ..... pentru folosirea datelor
implicite');
               writeln;
               writeln('      2 ..... pentru a introduce datele de
la tastatur†');
               writeln;
               readln(men1);
               writeln;
               writeln;
          end;

     procedure meniu2;
          begin
               writeln('    Tasteaz†:');
               writeln;
               writeln('            1 ..... Dati masa total† a
corpului');
               writeln;
               writeln('      2 ..... Dati masele celor 7 elemente
componente');
               writeln;
               readln(men2);
               writeln;
          end;

     procedure masa1;
          begin
               clrscr;
               writeln('Dati masa total† a corpului, exprimat† Œn
kg !');
               write('        m=');
               readln(m);
               writeln;
               m1:=7.3/100*m/2;
               m2:=15/100*m/2;
               m3:=19/100*m;
               m4:=9.7/100*m;
               m5:=36.5/100*m;
               m6:=7/100*m;
               m7:=5.5/100*m/2;
               m10:=m1;
               m9:=m2;
```

```
               m7:=m8;
               write('Doresti s† vezi valorile maselor componente?
(D/N)');
               readln(vm);
               writeln;
               if (vm='d') or (vm='D') then
                  begin
                       writeln;
                       writeln('Masele elementelor modelului
sunt:');
                       writeln;
                       writeln('     m1=',m1:9:3,' kg');
                       writeln('     m2=',m2:9:3,' kg');
                       writeln('     m3=',m3:9:3,' kg');
                       writeln('     m4=',m4:9:3,' kg');
                       writeln('     m5=',m5:9:3,' kg');
                       writeln('     m6=',m6:9:3,' kg');
                       writeln('     m7=',m7:9:3,' kg');
                       writeln('     m8=',m8:9:3,' kg');
                       writeln('     m9=',m9:9:3,' kg');
                       writeln('     m10=',m10:9:3,' kg');
                       tasta;
                  end;
          end;

     procedure masa2;
          begin
               clrscr;
               writeln('Dati masele elementelor componente ale
corpului, Œn kg !');
               writeln;
               write('      m1=');readln(m1);
               write('      m2=');readln(m2);
               write('      m3=');readln(m3);
               write('      m4=');readln(m4);
               write('      m5=');readln(m5);
               write('      m6=');readln(m6);
               write('      m7=');readln(m7);
               write('      m8=');readln(m8);
               write('      m9=');readln(m9);
               write('      m10=');readln(m10);
               writeln;
               m:=m1+m2+m3+m4+m5+m6+m7+m8+m9+m10;
               delay(850);
               writeln('Masa total† a corpului omenesc este:
m=',m:9:3,' kg');
               writeln;
               tasta1;
          end;

     procedure frecventa;
```

```
          begin
               writeln;
               writeln('Doresti s† dai valorile frecventelor de
rezonant† ale');
               write('    maselor componente ale modelului mecanic
? (D/N)');
               readln(vf);
               writeln;
               if (vf='d') or (vf='D') then
                  begin
                      writeln('Dati frecventele de rezonant†, Œn
Hz!');
                      writeln;
                      write('            fn1:=');readln(fn1);
                      write('            fn2:=');readln(fn2);
                      write('            fn3:=');readln(fn3);
                      write('            fn4:=');readln(fn4);
                      write('            fn5:=');readln(fn5);
                      write('            fn6:=');readln(fn6);
                      write('            fn7:=');readln(fn7);
                      write('            fn8:=');readln(fn8);
                      write('            fn9:=');readln(fn9);
                      write('            fn10:=');readln(fn10);
                      writeln;
                      write('Doresti s† vezi valorile pulsatiilor?
(D/N)');
                      readln(vp);
                      writeln;
                      if (vp='d') or (vp='D') then
                         begin
                              writeln;
                              writeln('Pulsatiile aferente
elementelor modelului sunt:');
                    writeln;
                    omega1:=2*pi*fn1;
                    writeln('     omega1=',omega1:7:2,' rad/s');
                    omega2:=2*pi*fn2;
                    writeln('     omega2=',omega2:7:2,' rad/s');
                    omega3:=2*pi*fn3;
                    writeln('     omega3=',omega3:7:2,' rad/s');
                    omega4:=2*pi*fn4;
                    writeln('     omega4=',omega4:7:2,' rad/s');
                    omega5:=2*pi*fn5;
                    writeln('     omega5=',omega5:7:2,' rad/s');
                    omega6:=2*pi*fn6;
                    writeln('     omega6=',omega6:7:2,' rad/s');
                    omega7:=2*pi*fn7;
                    writeln('     omega7=',omega7:7:2,' rad/s');
                    omega8:=2*pi*fn8;
                    writeln('     omega8=',omega8:7:2,' rad/s');
                    omega9:=2*pi*fn9;
                  writeln('     omega9=',omega9:7:2,' rad/s');
                  omega10:=2*pi*fn10;
                  writeln('     omega10=',omega10:7:2,' rad/s');
                  readln;
            end;
        end;
               if (vf<>'d') or (vf<>'D') then
                  begin
                        fn1:=8;  fn2:=6;  fn3:=8.5;  fn4:=3;
                        fn5:=8;  fn6:=5;  fn7:=14;
                        fn8:=14;  fn9:=6;  fn10:=8;
                  end;
          end;

     procedure date_init;
          begin
               m:=80;
               m1:=7.3/100*m/2;
               m2:=15/100*m/2;
               m3:=19/100*m;
               m4:=9.7/100*m;
               m5:=36.5/100*m;
               m6:=7/100*m;
               m7:=5.5/100*m/2;
               m8:=m7;
               m9:=m2;
               m10:=m1;
               fn1:=8;  fn2:=6;  fn3:=8.5;  fn4:=3;
               fn5:=8;  fn6:=5;  fn7:=14;
               fn8:=14;  fn9:=6;  fn10:=8;
             end;

     procedure calcul_afisare;
          begin
               clrscr;
               textbackground(1);
               textcolor(yellow);
               gotoxy(10,4);
               writeln('P A R A M E T R I I   M  O D E L U L U I
M E C A N I C:');
               gotoxy(10,5);
     writeln('ßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßß
ßßßßß');
               writeln;
               k12:=m2*sqr(2*pi*fn2)/2;
               k23:=k12;
               k:=m1*sqr(2*pi*fn1)-k12;
               k56:=m6*sqr(2*pi*fn6);
               k57:=m7*sqr(2*pi*fn7);
               k58:=m8*sqr(2*pi*fn8);
               k39:=m9*sqr(2*pi*fn9)/2;
               k910:=k39;
               a:=m3*sqr(2*pi*fn3)-k23-k39;
               b:=m4*sqr(2*pi*fn4);
               c:=m5*sqr(2*pi*fn5)-k56-k57-k58;
               k34:=(a+b-c)/2;
               k35:=(a-b+c)/2;
               k45:=(-a+b+c)/2;
               writeln('     Valorile   parametrilor   elementelor
elastice sunt:');
               writeln('     k12=',k12:10:2,' N/m');
               writeln('     k23=',k23:10:2,' N/m');
               writeln('     k34=',k34:10:2,' N/m');
               writeln('     k35=',k35:10:2,' N/m');
               writeln('     k45=',k45:10:2,' N/m');
               writeln('     k56=',k56:10:2,' N/m');
               writeln('     k57=',k57:10:2,' N/m');
               writeln('     k58=',k58:10:2,' N/m');
               writeln('     k39=',k39:10:2,' N/m');
               writeln('     k910=',k910:9:2,' N/m');
               writeln('     k=',k:12:2,' N/m');
               writeln; writeln;
               readln;
               m:=m1+m2+m3+m4+m5+m6+m7+m8+m9+m10;
               pr1:=m1*100/round(m);
               pr2:=m2*100/round(m);
               pr3:=m3*100/round(m);
               pr4:=m4*100/round(m);
               pr5:=m5*100/round(m);
               pr6:=m6*100/round(m);
               pr7:=m7*100/round(m);
               pr8:=m8*100/round(m);
               pr9:=m9*100/round(m);
               pr10:=m10*100/round(m);
               pr:=pr1+pr2+pr3+pr4+pr5+pr6+pr7+pr8+pr9+pr10;
               writeln('     Masele   elementelor   componente   ale
corpului sunt:');
               writeln('                                m1=',m1:9:2,'    kg
',pr1:5:2,'%');
               writeln('                                m2=',m2:9:2,'    kg
',pr2:5:2,'%');
               writeln('                                m3=',m3:9:2,'    kg
',pr3:5:2,'%');
               writeln('                                m4=',m4:9:2,'    kg
',pr4:5:2,'%');
               writeln('                                m5=',m5:9:2,'    kg
',pr5:5:2,'%');
               writeln('                                m6=',m6:9:2,'    kg
',pr6:5:2,'%');
               writeln('                                m7=',m7:9:2,'    kg
',pr7:5:2,'%');
```

```
               writeln('                        m8=',m8:9:2,'    kg
',pr8:5:2,'%');
               writeln('                        m9=',m9:9:2,'    kg
',pr9:5:2,'%');
               writeln('                       m10=',m10:9:2,'    kg
',pr10:5:2,'%');
               writeln;
               m:=m1+m2+m3+m4+m5+m6+m7+m8+m9+m10;
               writeln('   Masa  total†  a  corpului  omenesc  este:
m=',round(m),' kg');
               tasta1;
     end;

     procedure tiparire_imprimanta;
          begin
               clrscr;
               writeln('       Vrei  s†  tip†resti  rezultatele  la
imprimant† ?');
               writeln;
               writeln('Tasteaz†:');
               writeln;
               writeln('            1  .....  pentru  tip†rire  la
imprimant†');
               writeln;
               writeln('      2 ..... pentru NU');
               writeln;
               readln(vimp);
               writeln;
               if (vimp='1') then
                  begin
                     clrscr;
                     textbackground(7);
                     textcolor(5);
                     gotoxy(10,10);
                     writeln('Esti  sigur  c†  vrei  s†  tip†resti  la
imprimant† valorile');
                     write('              parametrilor ? (D/N)');
                     readln(cp);
                     writeln;
                     if (cp='d') or (cp='D') then
                       begin
                          textbackground(5);
                          clrscr;
                          textbackground(1);
                          textcolor(135);
                          gotoxy(30,10);
                          writeln('T I P A R I R E');
                          gotoxy(28,12);
                          writeln('*******************');
                          gotoxy(28,8);
                          writeln('*******************');
                          writeln(lst,'  P A R A M E T R I I   M
O D E L U L U I  M E C A N I C:');
                          writeln(lst,'
ßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßß');
                          writeln(lst,'');
                          writeln(lst,'     k12=',k12:9:2,' N/m');
                          writeln(lst,'     k23=',k23:9:2,' N/m');
                          writeln(lst,'     k34=',k34:9:2,' N/m');
                          writeln(lst,'     k35=',k35:9:2,' N/m');
                          writeln(lst,'     k45=',k45:9:2,' N/m');
                          writeln(lst,'     k56=',k56:9:2,' N/m');
                          writeln(lst,'     k57=',k57:9:2,' N/m');
                          writeln(lst,'     k58=',k58:9:2,' N/m');
                          writeln(lst,'     k39=',k39:9:2,' N/m');
                          writeln(lst,'          k910=',k910:8:2,'
N/m');
                          writeln(lst,'     k=',k:11:2,' N/m');
                          writeln(lst,''); writeln(lst,'');
                          writeln(lst,'       Masele    elementelor
componente ale corpului sunt:');
                          writeln(lst,'');
                          writeln(lst,'          m1=',m1:9:2,'  kg
',m1*100/round(m):5:1,' %');
                          writeln(lst,'          m2=',m2:9:2,'  kg
',m2*100/round(m):5:1,' %');
                          writeln(lst,'          m3=',m3:9:2,'  kg
',m3*100/round(m):5:1,' %');
                          writeln(lst,'          m4=',m4:9:2,'  kg
',m4*100/round(m):5:1,' %');
                          writeln(lst,'          m5=',m5:9:2,'  kg
',m5*100/round(m):5:1,' %');
                          writeln(lst,'          m6=',m6:9:2,'  kg
',m6*100/round(m):5:1,' %');
                          writeln(lst,'          m7=',m7:9:2,'  kg
',m7*100/round(m):5:1,' %');
                          writeln(lst,'          m8=',m8:9:2,'  kg
',m8*100/round(m):5:1,' %');
                          writeln(lst,'          m9=',m9:9:2,'  kg
',m9*100/round(m):5:1,' %');
                          writeln(lst,'         m10=',m10:9:2,'  kg
',m10*100/round(m):5:1,' %');
                          writeln(lst,'');
                          m:=m1+m2+m3+m4+m5+m6+m7+m8+m9+m10;
                          writeln(lst,'   Masa  total†  a  corpului
omenesc este: m=',round(m),' kg');
                          readln;
                          textcolor(7);
                       end;
                  end;
          end;
```

```
     procedure calcul;
         begin
               m1:=7.3/100*m/2;
               m2:=15/100*m/2;
               m3:=19/100*m;
               m4:=9.7/100*m;
               m5:=36.5/100*m;
               m6:=7/100*m;
               m7:=5.5/100*m/2;
               m8:=m7;
               m9:=m2;
               m10:=m1;
               fn1:=8;  fn2:=6;  fn3:=8.5;  fn4:=3;
               fn5:=8;  fn6:=5;  fn7:=14;
               fn8:=14;  fn9:=6;  fn10:=8;
               k12:=m2*sqr(2*pi*fn2)/2;
               k23:=k12;
               k:=m1*sqr(2*pi*fn1)-k12;
               k56:=m6*sqr(2*pi*fn6);
               k57:=m7*sqr(2*pi*fn7);
               k58:=m8*sqr(2*pi*fn8);
               k39:=m9*sqr(2*pi*fn9)/2;
               k910:=k39;
               a:=m3*sqr(2*pi*fn3)-k23-k39;
               b:=m4*sqr(2*pi*fn4);
               c:=m5*sqr(2*pi*fn5)-k56-k57-k58;
               k34:=(a+b-c)/2;
               k35:=(a-b+c)/2;
               k45:=(-a+b+c)/2;
         end;

     procedure param;
         begin
              clrscr;
              writeln; writeln; writeln;
              writeln('                   Variatia parametrilor kij
functie de masa totala');
              writeln('                ----------------------------
---------------------');
              writeln;
              writeln('
==================================================================
=========');
              writeln('     m[kg]   k12[N/m]    k23[N/m]   k34[N/m]
k35[N/m]  k45[N/m]  k56[N/m]  k57[N/m]');
              writeln('            k58[N/m]   k39[N/m]   k910[N/m]
k[N/m]');
              writeln('
==================================================================
=========');
              m:=50;
              while m<=100 do
                    begin
                       calcul;
                       writeln(' ',m:7:2,' ',k12:9:2,' ',k23:9:2,'
',k34:9:2,' ',k35:9:2,' ',k45:9:2,' ',k56:9:2,' ',k57:9:2);
                       writeln('           ',k58:9:2,' ',k39:9:2,'
',k910:9:2,' ',k:9:2);
                       m:=m+6;
                    end;
              writeln('
==================================================================
=========');
              readln;
         end;

     procedure param_im;
         begin
              writeln(lst);writeln(lst);
              writeln(lst,'              Variatia parametrilor kij
functie de masa totala');
              writeln(lst,'             ---------------------------
---------------------');
              writeln(lst);
              writeln(lst,'
==================================================================
=========');
              writeln(lst,'   m[kg]   k12[N/m]   k23[N/m]   k34[N/m]
k35[N/m]  k45[N/m]  k56[N/m]  k57[N/m]');
              writeln(lst,'          k58[N/m]   k39[N/m]   k910[N/m]
k[N/m]');
              writeln(lst,'
==================================================================
=========');
              m:=50;
              while m<=100 do
                    begin
                       calcul;
                       writeln(lst,'     ',m:7:2,'     ',k12:9:2,'
',k23:9:2,'   ',k34:9:2,'   ',k35:9:2,'   ',k45:9:2,'   ',k56:9:2,'
',k57:9:2);
                       writeln(lst,'                   ',k58:9:2,'
',k39:9:2,' ',k910:9:2,' ',k:9:2);
                       m:=m+6;
                    end;
         end;


     procedure param1;
         begin
              clrscr;
              writeln; writeln; writeln;
              writeln('                  Variatia maselor partilor
corpului functie de masa totala');
              writeln('             -------------------------------
--------------------------');
              writeln;
              writeln('
==================================================================
=========');
              writeln('    m[kg]      m1[kg]      m2[kg]      m3[kg]
m4[kg]    m5[kg]    m6[kg]    m7[kg]');
              writeln('     m8[kg]    m9[kg]    m10[kg]');
              writeln('
==================================================================
=========');


              m:=50;
              while m<=100 do
                    begin
                       calcul;
                       writeln('  ',m:7:2,'  ',m1:9:2,'  ',m2:9:2,'
',m3:9:2,' ',m4:9:2,' ',m5:9:2,' ',m6:9:2,' ',m7:9:2);
                       writeln('           ',m8:9:2,'   ',m9:9:2,'
',m10:9:2);
                       m:=m+6;
                    end;
              writeln('
==================================================================
=========');
              readln;
         end;

     procedure param_im1;
         begin
              writeln(lst);writeln(lst);
              writeln(lst,'              Variatia maselor partilor
corpului functie de masa totala');
              writeln(lst,'        ------------------------------
---------------------------');
              writeln(lst);
              writeln(lst,'   m[kg]     m1[kg]     m2[kg]     m3[kg]
m4[kg]    m5[kg]    m6[kg]    m7[kg]');
              writeln(lst,'     m8[kg]    m9[kg]    m10[kg]');
              writeln(lst,'
==================================================================
=========');
              m:=50;
              while m<=100 do
                    begin
                       calcul;
                       writeln(lst,'      ',m:7:2,'      ',m1:9:2,'
',m2:9:2,' ',m3:9:2,' ',m4:9:2,' ',m5:9:2,' ',m6:9:2,' ',m7:9:2);
                       writeln(lst,'        ',m8:9:2,'  ',m9:9:2,'
',m10:9:2);
                       m:=m+6;
                    end;
         end;


     begin
          antet;
          meniu1;
          if (men1=1) then
               date_init;
          if (men1=2) then
               begin
                    meniu2;
                    if (men2=1) then
                         masa1;
                    if (men2=2) then
                         masa2;
               end;
          frecventa;
          calcul_afisare;
          tiparire_imprimanta;
          clrscr;
          writeln;writeln;writeln;writeln;writeln;writeln;
          writeln(' Doriti sa consultati niste date statistice?');
          writeln;
          write('        [d/n]');
          readln(st);
          if (st='d') or (st='D') then
             begin
                  param;
                  writeln;writeln;
                  writeln(' Doriti sa le tipariti la
imprimanta?');
                  write('        [d/n]');
                  readln(im);
                  if (im='d') or (im='D') then
                     param_im;
                  param1;
                  writeln;writeln;
                  writeln(' Doriti sa le tipariti la
imprimanta?');
                  write('        [d/n]');
                  readln(im);
                  if (im='d') or (im='D') then
                     param_im1;
             end;
     end.
```

ANEXA 6.2

Codul sursã al programului *amortsim.pas*, scris pentru scrierea cãruia s-a folosit limbajul de programare Pascal, este:

```
program estimare_constante_elastice;
uses crt,printer;
var men1,men2:integer;
    m,s1,m1,m2,m3,m4,m5,m6,m7,m8,m9,m10,r:real;
    fn1,fn2,fn3,fn4,fn5,fn6,fn7,fn8,fn9,fn10,a,b,c:real;
    omega1,omega2,omega3,omega4,omega5,omega6,omega7,omega8,
omega9,omega10:real;
    k12,k23,k34,k35,k45,k56,k57,k58,k39,k910,k:real;
    p1,p2,p3,p4,p5,p6,p7,p8,p9,p10:real;
    n13,n34,n35,n45,n56,n57,n58,n310,c13,c34,c45,c56,c57,c310,c58:
real;
    c13cr,c34cr,c45cr,c56cr,c57cr,c45cr1,c310cr,c58cr:real;
    a3,b3,c3,v3,a4,b4,c4,v4,a5,b5,c5,v5:real;
    ch,vm,vp,vf,vimp,cp,ir:char;

procedure tasta1;
     begin
          ch:=readkey;
     end;

procedure antet;
     begin
          textbackground(3);
          clrscr;
          textbackground(1);
          textcolor(3);
          gotoxy(20,10);
          writeln('   M  O  D  E  L     M  E  C  A  N  I  C');
          gotoxy(10,12);
          writeln('al corpului omenesc asezat Œn picioare pe o
platform† vibrant†');
          writeln;
          delay(3050);
          clrscr;
          textbackground(3);
          textcolor(1);
          gotoxy(12,3);
          writeln('   E S T I M A R E   P A R A M E T R I   M O D
E L');
          writeln;
          writeln;
          tasta1;
     end;

procedure tasta;
```

```
     begin
          gotoxy(23,20);
          writeln('>> APASATI ORICE TASTA PENTRU A COMTINUA! <<');
          ch:=readkey;
     end;

procedure meniu1;
     begin
          writeln('       D A T E   I N I T I A L E');
          writeln('       ~~~~~~~~~~~~~~~~~~~~~~~~~');
          writeln('  Tasteaz†:');
          writeln;
          writeln('           1  .....  pentru  folosirea  datelor
implicite');
          writeln;
          writeln('      2 ..... pentru a introduce datele de la
tastatur†');
          writeln;
          readln(men1);
          writeln;
          writeln;
     end;

procedure meniu2;
     begin
          writeln('    Tasteaz†:');
          writeln;
          writeln('      1 ..... Dati masa total† a corpului');
          writeln;
          writeln('        2  .....  Dati  masele  celor  7  elemente
componente');
          writeln;
          readln(men2);
          writeln;
     end;

procedure masa1;
     begin
          clrscr;
          writeln;writeln;
          writeln;writeln;
          writeln('   Dati masa total† a corpului, exprimat† Œn kg
!');
          writeln;
          write('           m=');
          readln(m);
          m1:=7.3/100*m/2;
          m2:=15/100*m/2;
          m3:=19/100*m;
          m4:=9.7/100*m;
          m5:=36.5/100*m;
```

```
          m6:=7/100*m;
          m7:=5.5/100*m/2;
          m8:=m7;
          m9:=m2;
          m10:=m1;
          writeln;
          write('Doresti s† vezi valorile maselor componente?
(D/N)');
          readln(vm);
          writeln;
          if (vm='d') or (vm='D') then
             begin
                  writeln;
                  writeln('Masele elementelor modelului sunt:');
                  writeln;
                  writeln('     m1=',m1:9:3,' kg');
                  writeln('     m2=',m2:9:3,' kg');
                  writeln('     m3=',m3:9:3,' kg');
                  writeln('     m4=',m4:9:3,' kg');
                  writeln('     m5=',m5:9:3,' kg');
                  writeln('     m6=',m6:9:3,' kg');
                  writeln('     m7=',m7:9:3,' kg');
                  writeln('     m8=',m8:9:3,' kg');
                  writeln('     m9=',m9:9:3,' kg');
                  writeln('     m10=',m10:8:3,' kg');
                  tasta;
             end;
     end;

procedure masa2;
     begin
          clrscr;
          writeln('Dati masele elementelor componente ale
corpului, Œn kg !');
          writeln;
          write('      m1=');readln(m1);
          write('      m2=');readln(m2);
          write('      m3=');readln(m3);
          write('      m4=');readln(m4);
          write('      m5=');readln(m5);
          write('      m6=');readln(m6);
          write('      m7=');readln(m7);
          write('      m8=');readln(m8);
          write('      m9=');readln(m9);
          write('      m10=');readln(m10);
          writeln;
          m:=m1+m2+m3+m4+m5+m6+m7+m8+m9+m10;
          delay(850);
          writeln('Masa total† a corpului omenesc este:
m=',m:9:2,' kg');
          writeln;
          tasta1;
     end;

procedure amortizare;
     begin
          clrscr;
          writeln;writeln;writeln;
          writeln('Dati valoarea procentual† a raportului dintre
amortizarea');
          writeln('  real† si cea critic† pentru fiecare element
cu amortizare');
          writeln('  vƒscoast† Œn parte !');
          writeln;
          write('      n13=');readln(n13);
          write('      n34=');readln(n34);
          write('      n45=');readln(n45);
          write('      n56=');readln(n56);
          write('      n57=');readln(n57);
          write('      n58=');readln(n58);
          write('     n310=');readln(n310);
          writeln;
          n13:=n13/100;n34:=n34/100;
          n45:=n45/100;n56:=n56/100;n57:=n57/100;
          n58:=n58/100;n310:=n310/100;
          tasta1;
     end;

procedure frecventa;
     begin
          writeln;
          writeln('Doresti s† dai valorile frecventelor de
rezonant† ale');
          write('     maselor componente ale modelului mecanic ?
(D/N)');
          readln(vf);
          writeln;
          if (vf='d') or (vf='D') then
             begin
                 writeln('Dati frecventele de rezonant†, Œn Hz!');
                 writeln;
                 write('             fn1=');readln(fn1);
                 write('             fn2=');readln(fn2);
                 write('             fn3=');readln(fn3);
                 write('             fn4=');readln(fn4);
                 write('             fn5=');readln(fn5);
                 write('             fn6=');readln(fn6);
                 write('             fn7=');readln(fn7);
                 write('             fn8=');readln(fn8);
                 write('             fn9=');readln(fn9);
                 write('            fn10=');readln(fn10);
                 writeln;
                 p1:=2*pi*fn1;p2:=2*pi*fn2;p3:=2*pi*fn3;
                 p4:=2*pi*fn4;p5:=2*pi*fn5;p6:=2*pi*fn6;
                 p7:=2*pi*fn7;
                 p8:=2*pi*fn8;p9:=2*pi*fn9;p10:=2*pi*fn10;
                 write('Doresti   s†   vezi   valorile   pulsatiilor?
(D/N)');
                 readln(vp);
                 writeln;
                 if (vp='d') or (vp='D') then
                    begin
                         writeln;
                   writeln('Pulsatiile     aferente     elementelor
modelului sunt:');
                         writeln;
                         writeln('     p1=',p1:7:2,' rad/s');
                         writeln('     p2=',p2:7:2,' rad/s');
                         writeln('     p3=',p3:7:2,' rad/s');
                         writeln('     p4=',p4:7:2,' rad/s');
                         writeln('     p5=',p5:7:2,' rad/s');
                         writeln('     p6=',p6:7:2,' rad/s');
                         writeln('     p7=',p7:7:2,' rad/s');
                         writeln('     p8=',p8:7:2,' rad/s');
                         writeln('     p9=',p9:7:2,' rad/s');
                         writeln('    p10=',p10:7:2,' rad/s');
                         readln;
                    end;
             end;
          if (vf<>'d') or (vf<>'D') then
             begin
                   fn1:=8;  fn2:=6;  fn3:=8.5;  fn4:=3;
                   fn5:=8;  fn6:=5;  fn7:=14;
                   fn8:=14;  fn9:=6;  fn10:=8;
                   p1:=2*pi*fn1;p2:=2*pi*fn2;p3:=2*pi*fn3;
                   p4:=2*pi*fn4;
                   p5:=2*pi*fn5;p6:=2*pi*fn6;p7:=2*pi*fn7;
                   p8:=2*pi*fn8;p9:=2*pi*fn9;p10:=2*pi*fn10;
              end;
     end;

procedure date_init;
     begin
          m:=80;
          m1:=7.3/100*m/2;
          m2:=15/100*m/2;
          m3:=19/100*m;
          m4:=9.7/100*m;
          m5:=36.5/100*m;
          m6:=7/100*m;
          m7:=5.5/100*m/2;
          m8:=m7;
          m9:=m2;
          m10:=m1;
          fn1:=8;  fn2:=6;  fn3:=8.5;  fn4:=3;
          fn5:=8;  fn6:=5;  fn7:=14;
          fn8:=14;  fn9:=6;  fn10:=8;
          p1:=2*pi*fn1;p2:=2*pi*fn2;p3:=2*pi*fn3;p4:=2*pi*fn4;
          p5:=2*pi*fn5;p6:=2*pi*fn6;p7:=2*pi*fn7;
          p8:=2*pi*fn8;p9:=2*pi*fn9;p10:=2*pi*fn10;
          n13:=0.05;n34:=0.10;n45:=0.25;n56:=0.15;n57:=0.10;
          n58:=0.10;n310:=0.05;
      end;

procedure calcul;
     begin
       k12:=m2*p2*p2/2;
       k23:=k12;
       k:=m1*p1*p1/(1-n13*n13)-k12;
       c13cr:=2*sqrt((k12+k)*m1);
       c13:=n13*c13cr;
       k39:=m9*p9*p9/2;
       k910:=k39;
       c310cr:=c13cr;
       c310:=c13;
       a3:=1-sqr(n34);
       b3:=2*n34*(n34-n13)*sqrt(m1*k12);
       c3:=-(sqr(m3*p3)+sqr(n13-n34)*m1*k12);
       v3:=(-b3+sqrt(b3*b3-4*a3*c3))/(2*a3);
       a:=v3*v3/m3-k23-k39;
       c34cr:=2*(sqrt(m3*(k23+a))-sqrt(m1*k12));
       c34:=n34*c34cr;
       a4:=1-sqr(n45);
       b4:=n45*(n45-n34)*c34cr;
       c4:=-sqr(m4*p4)-sqr(n45-n34)*sqr(c34cr)/4;
       v4:=(-b4+sqrt(b4*b4-4*a4*c4))/(2*a4);
       b:=v4*v4/m4;
       c45cr:=-(2*sqrt(m4*b)-c34cr);
       c45:=n45*c45cr;
       k56:=m6*p6*p6/(1-sqr(n56));
       c56cr:=2*sqrt(k56*m6);
       c56:=n56*c56cr;
       k57:=m7*p7*p7/(1-sqr(n57));
       c57cr:=2*sqrt(k57*m7);
       c57:=n57*c57cr;
       k58:=m8*p8*p8/(1-sqr(n58));
       c58cr:=2*sqrt(k58*m8);
       c58:=n58*c58cr;
       b5:=n45*(c56cr+c57cr-c56-c57);
       c5:=-(sqr(m5*p5)+sqr(c56cr+c57cr-c56-c57));
       v5:=(-b5+sqrt(b5*b5-4*a5*c5))/(2*a5);
       c:=v5*v5/m5-k56-k57;
       c:=m5*sqr(p5)+(c45+c56+c57)/(4*m5)-k56-k57-k58;
       k34:=(a+b-c)/2;
       k35:=(a-b+c)/2;
       k45:=(-a+b+c)/2;
       {writeln('c45cr=',c45cr:9:2);
       writeln('c34cr=',c34cr:9:2);
       writeln('c56cr=',c56cr:9:2);
       writeln('c57cr=',c57cr:9:2);
       writeln('c13cr=',c13cr:9:2);}
       readln;
     end;

procedure afisare;
     begin
          clrscr;
          textbackground(1);
          textcolor(yellow);
          gotoxy(10,4);
          writeln('P A R A M E T R I I   M  O D E L U L U I  M E C
A N I C:');
          gotoxy(10,5);
writeln('ßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßß
ßßßßß');
          writeln;
          writeln;
          writeln('     k12=',k12:9:2);
          writeln('     k23=',k23:9:2);
          writeln('     k34=',k34:9:2);
          writeln('     k35=',k35:9:2);
          writeln('     k45=',k45:9:2);
          writeln('     k56=',k56:9:2);
          writeln('     k57=',k57:9:2);
          writeln('     k58=',k58:9:2);
          writeln('     k39=',k39:9:2);
          writeln('     k910=',k910:8:2);
          writeln('     k=',k:11:2);
          writeln;
          writeln('     c13=',c13:9:2);
          writeln('     c34=',c34:9:2);
          writeln('     c45=',c45:9:2);
          writeln('     c56=',c56:9:2);
          writeln('     c57=',c57:9:2);
          writeln('     c58=',c58:9:2);
          writeln('     c310=',c310:8:2);
          readln;
     end;

procedure tiparire_imprimanta;
     begin
          clrscr;
          writeln('    Vrei st tiptresti rezultatele la imprimantt
?');
          writeln;
          writeln('Tasteaz†:');
          writeln;
          writeln('      1 ..... pentru tip†rire la imprimant†');
          writeln;
          writeln('      2 ..... pentru NU');
          writeln;
          readln(vimp);
          writeln;
          if (vimp='1') then
             begin
                clrscr;
                textbackground(7);
                textcolor(5);
                gotoxy(10,10);
           writeln('Esti sigur c† vrei s† tip†resti la imprimant†
valorile');
                write('              parametrilor ? (D/N)');
                readln(cp);
                writeln;
                if (cp='d') or (cp='D') then
                  begin
                     writeln('Doresti     s†     tip†resti     la
imprimant†:');
                     writeln;
                     writeln('     1 ..... numai rezultatele');
                     writeln;
                     writeln('     2 ..... si datele de intrare');
                     writeln;
                     readln(ir);
                     textbackground(5);
                     clrscr;
                     textbackground(1);
                     textcolor(135);
                     gotoxy(30,10);
                     writeln('T I P A R I R E');
                     gotoxy(30,12);
                     writeln('***************');
                     if ir='2' then begin
                        writeln(lst,'Datele de intrare!');
                        writeln(lst,'~~~~~~~~~~~~~~~~~~');
                        writeln(lst,'');
                        writeln(lst,'   fn1=',fn1:7:2,' Hz');
                        writeln(lst,'   fn2=',fn2:7:2,' Hz');
                        writeln(lst,'   fn3=',fn3:7:2,' Hz');
                        writeln(lst,'   fn4=',fn4:7:2,' Hz');
                        writeln(lst,'   fn5=',fn5:7:2,' Hz');
                        writeln(lst,'   fn6=',fn6:7:2,' Hz');
                        writeln(lst,'   fn7=',fn7:7:2,' Hz');
                        writeln(lst,'   fn8=',fn8:7:2,' Hz');
                        writeln(lst,'   fn9=',fn9:7:2,' Hz');
                        writeln(lst,'   fn10=',fn10:7:2,' Hz');
```

```
                       writeln(lst,'');writeln(lst,'');
                       writeln(lst,'   n13=',n13*100:6:2,' %');
                       writeln(lst,'   n34=',n34*100:6:2,' %');
                       writeln(lst,'   n45=',n45*100:6:2,' %');
                       writeln(lst,'   n56=',n56*100:6:2,' %');
                       writeln(lst,'   n57=',n57*100:6:2,' %');
                       writeln(lst,'   n58=',n58*100:6:2,' %');
                       writeln(lst,'   n310=',n310*100:6:2,' %');
                       writeln(lst,'');writeln(lst,'');
                    end;
      writeln(lst,'P A R A M E T R I I   M O D E L U L U I  M E C
A N I C:');
      writeln(lst,'ßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßß
ßßßßßßßßßß');
                    writeln(lst,'');
                    writeln(lst,'     k12=',k12:9:2);
                    writeln(lst,'     k23=',k23:9:2);
                    writeln(lst,'     k34=',k34:9:2);
                    writeln(lst,'     k35=',k35:9:2);
                    writeln(lst,'     k45=',k45:9:2);
                    writeln(lst,'     k56=',k56:9:2);
                    writeln(lst,'     k57=',k57:9:2);
                    writeln(lst,'     k58=',k58:9:2);
                    writeln(lst,'     k39=',k39:9:2);
                    writeln(lst,'     k910=',k910:9:2);
                    writeln(lst,'     k=',k:9:2);
                    writeln(lst,''); writeln(lst,'');
                    writeln(lst,'Masele   elementelor   componente
ale corpului sunt:');
                    writeln(lst,'*****************************
*****************');
                    writeln(lst,'');}
                    writeln(lst,'      m1=',m1:9:3,' kg');
                    writeln(lst,'      m2=',m2:9:3,' kg');
                    writeln(lst,'      m3=',m3:9:3,' kg');
                    writeln(lst,'      m4=',m4:9:3,' kg');
                    writeln(lst,'      m5=',m5:9:3,' kg');
                    writeln(lst,'      m6=',m6:9:3,' kg');
                    writeln(lst,'      m7=',m7:9:3,' kg');
                    writeln(lst,'      m8=',m8:9:3,' kg');
                    writeln(lst,'      m9=',m9:9:3,' kg');
                    writeln(lst,'      m10=',m10:9:3,' kg');
                    writeln(lst,'');
                    m:=m1+m2+m3+m4+m5+m6+m7+m8+m9+m10;
                    writeln(lst,'Masa  total†  a  corpului  omenesc
este: m=',m:7:3,' kg');
                    writeln(lst,'');writeln(lst,'');
                    writeln(lst,'     c13=',c13:9:2);
                    writeln(lst,'     c34=',c34:9:2);
                    writeln(lst,'     c45=',c45:9:2);
                    writeln(lst,'     c56=',c56:9:2);
```

```
                     writeln(lst,'     c57=',c57:9:2);
                     writeln(lst,'     c58=',c58:9:2);
                     writeln(lst,'     c310=',c310:9:2);
                     writeln;
                  end;
             end;
     end;

begin
     antet;
     meniu1;
     if (men1=1) then
          date_init;
     if (men1=2) then
          begin
               meniu2;
               if (men2=1) then
                    masa1;
               if (men2=2) then
                    masa2;
               amortizare;
          end;
     frecventa;
     calcul;
     afisare;
     tiparire_imprimanta;
end.
```

ANEXA 6.3

Conținutul fişierului de comenzi indirecte *paramsim.bat* este:

```
ECHO OFF
:RELUARE
CLS
ECHO
ECHO --------------------------------------------------------------
---
ECHO !               K : fără amortizare
ECHO !               C : cu amortizare
ECHO !               X : EXIT
ECHO --------------------------------------------------------------
---
ECHO
CHOICE /c:KCX Alegeti o comandă
IF ERRORLEVEL 3 GOTO FINAL
IF ERRORLEVEL 2 GOTO AM
IF ERRORLEVEL 1 GOTO ARC

:ARC
ECHO --------------------------------------------------------------
---
ECHO !              A: programul e pe dischetă
ECHO !              C: programul e pe harddisc
ECHO --------------------------------------------------------------
---
ECHO
CHOICE /c:AC Optiunea:
IF ERRORLEVEL 2 GOTO ARC1
IF ERRORLEVEL 1 GOTO ARC2

:ARC2
CALL A:\ELASTSIM.EXE
PAUSE
GOTO RELUARE

:ARC1
CALL C:\LUCRU\ELASTSIM.EXE
PAUSE
GOTO RELUARE

:AM
ECHO --------------------------------------------------------------
---
ECHO !              A: programul e pe dischetă
ECHO !              C: programul e pe harddisk
ECHO --------------------------------------------------------------
---
ECHO
CHOICE /c:AC Optiunea:
IF ERRORLEVEL 2 GOTO AM1
IF ERRORLEVEL 1 GOTO AM2

:AM2
CALL A:\AMORTSIM.EXE
GOTO RELUARE

:AM1
CALL C:\LUCRU\AMORTSIM.EXE
GOTO RELUARE

:FINAL
ECHO
```

ANEXA 6.4

Codul sursă al programului *kcm.m* pentru simularea comportării la vibrații a modelului mecanic al corpului omenesc, model reprezentat în fig.6.1 este redat în continuare:

```
%Program pentru simularea functionarii unui model mecanic
%al corpului omenesc
%stergere ecran in mod text
cla
%stergere ecran in mod grafic
clg
%stergerea variabilelor existente in memorie
clear
disp('   M O D E L    M E C A N I C');
disp('   ~~~~~~~~~    ~~~~~~~~~~~~~');
format compact;
i=sqrt(-1);
%introducerea datelor initiale:
t1=0;
while t1==0
    k1=menu('INITIALIZARI','de la TASTATURA','date
implicite','continue');
    if k1==1
     disp('Introduceti datele initiale ale problemei !');
     k12=input('k12=');
     k23=input('k23=');
     k34=input('k34=');
     k45=input('k45=');
     k56=input('k56=');
     k57=input('k57=');
     k58=input('k58=');
     k39=input('k39=');
     k910=input('k910=');
     k=input('k=');

        k21=menu('TIPUL MODELULUI','cu amortizare','fara
amortizare');
        if k21==1

          c13=input('c13=');
          c34=input('c34=');
          c45=input('c45=');
          c56=input('c56=');
          c57=input('c57=');
          c58=input('c58=');
          c310=input('c310=');

        end;                  %ifk21=1
```

```
         if k21==2
            c13=0;c34=0;c45=0;c56=0;c57=0;c58=0;c310=0;
         end;                 %ifk21=2

     t6=0;
     while t6==0
     k6=menu('masa corpului','toate masele componente','doar masa
corpului');
     if k6==1
         m1=input('m1=');
         m2=input('m2=');
         m3=input('m3=');
         m4=input('m4=');
         m5=input('m5=');
         m6=input('m6=');
         m7=input('m7=');
         m8=input('m8=');
         m9=input('m9=');
         m10=input('m10=');
%t1=1;
         t6=1;
     end;                  %ifk6=1
     if k6==2
         m=input('m=');
         m1=6.82*m/100/2;
         m2=8.52*m/100/2;
         m3=40.91*m/100;
         m4=1.71*m/100;
         m5=0.57*m/100;
         m6=7.38*m/100;
         m7=34.09*m/100/2;
         m8=m7;
         m9=m2;
         m10=m1;

         t6=1;
     end;            %ifk6=2
     end;            %while_t6
    end;                  %ifk1=1
    if k1==2;

     m1=5.44/2;
     m2=6.82/2;
     m3=32.72;
     m4=1.36;
     m5=0.46;
     m6=5.91;
     m7=27.22/2;
     m8=m7;
     m9=m2;
     m10=m1;
```

```
         k20=menu(' ','cu amortizare','fara amortizare');
        if k20==1
      k12=11843.53;
      k23=11843.53;
      k34=1276.75;
      k35=18900.4;
      k45=1480.7;
      k56=10942.3;
      k57=5089.01;
      k58=5089.01;
      k39=11843.53;
      k910=11843.53;
      k=14355.79;
      c13=55.32;
      c34=112.57;
      c45=109.62;
      c56=49.51;
      c57=42.32;
      c58=42.32;
      c310=55.32;
         end;                 %ifk20=1

        if k20==2
      k12=11843.53;
      k23=11843.53;
      k34=764.7;
      k35=18903.45;
      k45=1992.48;
      k56=10832.88;
      k57=4885.45;
      k58=4885.45;
      k39=11843.53;
      k910=11843.53;
      k=14093.8;
      c13=0;c34=0;c45=0;c56=0;c57=0;c27=0;c58=0;c310=0;
         end;                 %ifk20=2

    end;                       %ifk1=2

    if k1==3
     t1=1;
    end;                       %ifk1=3
end;                           %while t1
%definirea numaratorilor si numitorilor functiilor de transfer
%corespunzatoare blocurilor componente
n1=1;
d1=1;
n2=k12;
d2=1;
n3=1;
```

```
d3=[m1 0];
n4=c13;
d4=1;
n5=c13;
d5=1;
n6=1;
d6=[1 0];
n7=k12;
d7=1;
n8=k12+k23;
d8=1;
n9=1;
d9=[m2 0 0];
n10=k23;
d10=1;
n11=k23;
d11=1;
n12=k35;
d12=1;
n13=k34;
d13=1;
n14=k23+k34+k35+k39;
d14=1;
n15=1;
d15=[m3 0];
n16=c13+c34+c310;
d16=1;
n17=c34;
d17=1;
n18=1;
d18=[1 0];
n19=k34;
d19=1;
n20=c34;
d20=1;
n21=k45;
d21=1;
n22=k34+k45;
d22=1;
n23=1;
d23=[m4 0];
n24=c34+c45;
d24=1;
n25=c45;
d25=1;
n26=1;
d26=[1 0];
n27=k35;
d27=1;
n28=k45;
d28=1;
```

```
n29=c45;
d29=1;
n30=k56;
d30=1;
n31=k35+k45+k56+k57+k58;
d31=1;
n32=1;
d32=[m5 0];
n33=c45+c56+c57+c58;
d33=1;
n34=c56;
d34=1;
n35=c57;
d35=1;
n36=1;
d36=[1 0];
n37=k57;
d37=1;
n38=k56;
d38=1;
n39=k57;
d39=1;
n40=c57;
d40=1;
n41=1;
d41=[m6 0];
n42=c56;
d42=1;
n43=1;
d43=[1 0];
n44=1;
d44=[m7 0];
n45=1;
d45=[1 0];
n46=1;
d46=1;
n47=k910;
d47=1;
n48=1;
d48=[m10 0];
n49=c310;
d49=1;
n50=c310;
d50=1;
n51=1;
d51=[1 0];
n52=k910;
d52=1;
n53=k910+k39;
d53=1;
n54=1;
```

```
d54=[m9 0];
n55=1;
d55=[1 0];
n56=k39;
d56=1;
n57=k39;
d57=1;
n58=c58;
d58=1;
n59=k58;
d59=1;
n60=k58;
d60=1;
n61=c58;
d61=1;
n62=1;
d62=[m8 0];
n63=1;
d63=[1 0];
n64=k;
d64=1;
n65=k;
d65=1;

%definirea numarului de blocuri
nblocks=65;
blkbuild;
%specificarea interconexiunilor dintre blocuri
q=[1 0 0 0 0 0 0 0 0 0 0 0
   2 9 0 0 0 0 0 0 0 0 0 0
   3 2 -4 5 -7 64 0 0 0 0 0 0
   4 3 0 0 0 0 0 0 0 0 0 0
   5 15 0 0 0 0 0 0 0 0 0 0
   6 3 0 0 0 0 0 0 0 0 0 0
   7 6 0 0 0 0 0 0 0 0 0 0
   8 9 0 0 0 0 0 0 0 0 0 0
   9 7 -8 0 0 0 0 0 0 0 0 0
   10 18 0 0 0 0 0 0 0 0 0 0
   11 9 0 0 0 0 0 0 0 0 0 0
   12 36 0 0 0 0 0 0 0 0 0 0
   13 26 0 0 0 0 0 0 0 0 0 0
   14 18 0 0 0 0 0 0 0 0 0 0
   15 4 11 12 13 -14 -16 17 49 57 0 0
   16 15 0 0 0 0 0 0 0 0 0 0
   17 23 0 0 0 0 0 0 0 0 0 0
   18 15 0 0 0 0 0 0 0 0 0 0
   19 18 0 0 0 0 0 0 0 0 0 0
   20 15 0 0 0 0 0 0 0 0 0 0
   21 36 0 0 0 0 0 0 0 0 0 0
   22 26 0 0 0 0 0 0 0 0 0 0
   23 19 20 21 -22 -24 25 0 0 0 0 0
   24 23 0 0 0 0 0 0 0 0 0 0
   25 32 0 0 0 0 0 0 0 0 0 0
   26 23 0 0 0 0 0 0 0 0 0 0
   27 18 0 0 0 0 0 0 0 0 0 0
   28 26 0 0 0 0 0 0 0 0 0 0
   29 23 0 0 0 0 0 0 0 0 0 0
   30 43 0 0 0 0 0 0 0 0 0 0
   31 36 0 0 0 0 0 0 0 0 0 0
   32 27 28 29 30 -31 -33 34 35 37 58 59
   33 32 0 0 0 0 0 0 0 0 0 0
   34 41 0 0 0 0 0 0 0 0 0 0
   35 44 0 0 0 0 0 0 0 0 0 0
   36 32 0 0 0 0 0 0 0 0 0 0
   37 45 0 0 0 0 0 0 0 0 0 0
   38 36 -43 0 0 0 0 0 0 0 0 0
   39 36 -45 0 0 0 0 0 0 0 0 0
   40 32 -44 0 0 0 0 0 0 0 0 0
   41 38 42 0 0 0 0 0 0 0 0 0
   42 32 -41 0 0 0 0 0 0 0 0 0
   43 41 0 0 0 0 0 0 0 0 0 0
   44 39 40 0 0 0 0 0 0 0 0 0
   45 44 0 0 0 0 0 0 0 0 0 0
   46 0 0 0 0 0 0 0 0 0 0 0
   47 55 0 0 0 0 0 0 0 0 0 0
   48 47 -49 50 -52 65 0 0 0 0 0 0
   49 48 0 0 0 0 0 0 0 0 0 0
   50 15 0 0 0 0 0 0 0 0 0 0
   51 48 0 0 0 0 0 0 0 0 0 0
   52 51 0 0 0 0 0 0 0 0 0 0
   53 55 0 0 0 0 0 0 0 0 0 0
   54 52 -53 56 0 0 0 0 0 0 0 0
   55 54 0 0 0 0 0 0 0 0 0 0
   56 18 0 0 0 0 0 0 0 0 0 0
   57 55 0 0 0 0 0 0 0 0 0 0
   58 62 0 0 0 0 0 0 0 0 0 0
   59 63 0 0 0 0 0 0 0 0 0 0
   60 36 -63 0 0 0 0 0 0 0 0 0
   61 32 -62 0 0 0 0 0 0 0 0 0
   62 60 61 0 0 0 0 0 0 0 0 0
   63 62 0 0 0 0 0 0 0 0 0 0
   64 1 -6 0 0 0 0 0 0 0 0 0
   65 46 -51 0 0 0 0 0 0 0 0 0];

%specificarea intrarilor si iesirilor
iu=[1 46];
iy=[6 9 18 26 36 43 45 63 55 51];

iu1=1;iu2=46;

%interconectarea propriu-zisa
[A B C D]=connect(a,b,c,d,q,iu,iy);
```

```
%realizarea unui model minimal
[Am Bm Cm Dm]=minreal(A,B,C,D);
%prin MM-ISI avand matricele Am,Bm,Cm,Dm

iy1=6;iu1=1;
[A1 B1 C1 D1]=connect(a,b,c,d,q,iu,iy1);
[Am1 Bm1 Cm1 Dm1]=minreal(A1,B1,C1,D1);

iy2=9;iu1=1;
[A2 B2 C2 D2]=connect(a,b,c,d,q,iu,iy2);
[Am2 Bm2 Cm2 Dm2]=minreal(A2,B2,C2,D2);

iy3=18;iu1=1;
[A3 B3 C3 D3]=connect(a,b,c,d,q,iu,iy3);
[Am3 Bm3 Cm3 Dm3]=minreal(A3,B3,C3,D3);

iy4=26;
[A4 B4 C4 D4]=connect(a,b,c,d,q,iu,iy4);
[Am4 Bm4 Cm4 Dm4]=minreal(A4,B4,C4,D4);

iy5=36;
[A5 B5 C5 D5]=connect(a,b,c,d,q,iu,iy5);
[Am5 Bm5 Cm5 Dm5]=minreal(A5,B5,C5,D5);

iy6=43;
[A6 B6 C6 D6]=connect(a,b,c,d,q,iu,iy6);
[Am6 Bm6 Cm6 Dm6]=minreal(A6,B6,C6,D6);

iy7=45;
[A7 B7 C7 D7]=connect(a,b,c,d,q,iu,iy7);
[Am7 Bm7 Cm7 Dm7]=minreal(A7,B7,C7,D7);

iy8=63;
[A8 B8 C8 D8]=connect(a,b,c,d,q,iu,iy8);
[Am8 Bm8 Cm8 Dm8]=minreal(A8,B8,C8,D8);

iy9=55;
[A9 B9 C9 D9]=connect(a,b,c,d,q,iu,iy9);
[Am9 Bm9 Cm9 Dm9]=minreal(A9,B9,C9,D9);

iy10=51;
[A10 B10 C10 D10]=connect(a,b,c,d,q,iu,iy10);
[Am10 Bm10 Cm10 Dm10]=minreal(A10,B10,C10,D10);


[num1 den1]=ss2tf(Am1,Bm1,Cm1,Dm1,iu1);
%pause;
[z1,p1,k]=tf2zp(num1,den1);

%[num1p den1p]=ss2tf(Am1,Bm1,Cm1,Dm1,iu2);
%pause;
```

```
%[z1p,p1p,k]=tf2zp(num1p,den1p);

[num2 den2]=ss2tf(Am2,Bm2,Cm2,Dm2,iu1);
%pause;
[z2,p2,k]=tf2zp(num2,den2);

[num3 den3]=ss2tf(Am3,Bm3,Cm3,Dm3,iu1);
%pause;
[z3,p3,k]=tf2zp(num3,den3);

[num4 den4]=ss2tf(Am4,Bm4,Cm4,Dm4,iu1);
%pause;
[z4,p4,k]=tf2zp(num4,den4);

[num5 den5]=ss2tf(Am5,Bm5,Cm5,Dm5,iu1);
%pause;
[z5,p5,k]=tf2zp(num5,den5);

[num6 den6]=ss2tf(Am6,Bm6,Cm6,Dm6,iu1);
%pause;
[z6,p6,k]=tf2zp(num6,den6);
%pause

[num7 den7]=ss2tf(Am7,Bm7,Cm7,Dm7,iu1);
%pause;
[z7,p7,k]=tf2zp(num7,den7);
%pause

[num8 den8]=ss2tf(Am8,Bm8,Cm8,Dm8,iu1);
%pause;
[z8,p8,k]=tf2zp(num8,den8);

[num9 den9]=ss2tf(Am9,Bm9,Cm9,Dm9,iu1);
%pause;
[z9,p9,k]=tf2zp(num9,den9);
%pause

[num10 den10]=ss2tf(Am10,Bm10,Cm10,Dm10,iu1);
%pause;
[z10,p10,k]=tf2zp(num10,den10);
%pause

Am1;
%pause
Bm1;
%pause
Cm1;
%pause
Dm1;
%pause
```

```
w=logspace(-1,3,100);

t11=0;
while t11==0

k11=menu('CARACTERISTICI_FRECVENTA','H1','H2','H3','H4','H5','H6',
'H7','H8','H9','H10','PE_RAND','SUPRAPUSE','CONTINUE');
%Hi=functii de transfer=zi/u

  if k11==1
     clg;
    if (real(p1)<=0.00001),disp('Sistem stabil')
        else, disp(''),disp('    Pentru H1(s) exista poli
nenegativi')
        disp('                   ==> sistemul este instabil !!!');
    end             %if
   [re1,im1]=nyquist(num1,den1,w);
   [mod1,arg1]=bode(num1,den1,w);
   modd1=20*log10(mod1);
   semilogx(w,modd1,'w'),grid,title('c.a.p.1'),xlabel('pulsatia'),
ylabel('Ampl 1');
   pause
   if (real(p1)<0),loglog(w,modd1),grid;
       title('c.a.p.1'),xlabel('pulsatia'),ylabel('Ampl 1');
    end;               %if
    semilogx(w,arg1),grid,title('c.f.p.1'),xlabel('pulsatia'),
ylabel('Arg 1');
    pause
end;                    %ifk11=1

if k11==2
  if (real(p2)<=0.00001),disp('Sistem stabil')
     else, disp(''),disp('    Pentru H2(s) exista poli
nenegativi')
     disp('                   ==> sistemul este instabil !!!');
end             %if
[re2,im2]=nyquist(num2,den2,w);
[mod2,arg2]=bode(num2,den2,w);
modd2=20*log10(mod2);
semilogx(w,modd2,'w'),grid,title('c.a.p.2'),xlabel('pulsatia'),yla
bel('Ampl 2');
pause
if (real(p2)<0),loglog(w,modd2),grid;
  title('c.a.p.2'),xlabel('pulsatia'),ylabel('Ampl 2');
end;               %if
semilogx(w,arg2),grid,title('c.f.p.2'),xlabel('pulsatia'),ylabel('
Arg 2');
pause
end                  %ifk11=2

if k11==3
```

```
  if (real(p3)<=0.00001),disp('Sistem stabil')
     else, disp(''),disp('    Pentru H3(s) exista poli
nenegativi')
     disp('                  ==> sistemul este instabil !!!');
end             %if
[re3,im3]=nyquist(num3,den3,w);
[mod3,arg3]=bode(num3,den3,w);
modd3=20*log10(mod3);
semilogx(w,modd3,'w'),grid,title('c.a.p.3'),xlabel('pulsatia'),yla
bel('Ampl 3');
pause
if (real(p3)<0),loglog(w,modd3),grid;
  title('c.a.p.3'),xlabel('pulsatia'),ylabel('Ampl 3');
end;                %if
semilogx(w,arg3),grid,title('c.f.p.3'),xlabel('pulsatia'),ylabel('
Arg 3');
pause

end                   %ifk11=3

if k11==4

if (real(p4)<=0.00001),disp('Sistem stabil')
     else, disp(''),disp('    Pentru H4(s) exista poli
nenegativi')
     disp('                  ==> sistemul este instabil !!!');
end             %if
 [re4,im4]=nyquist(num4,den4,w);
 [mod4,arg4]=bode(num4,den4,w);
modd4=20*log10(mod4);
semilogx(w,modd4,'w'),grid,title('c.a.p.4'),xlabel('pulsatia'),yla
bel('Ampl 4');
pause
if (real(p4)<0),loglog(w,modd4),grid;
  title('c.a.p.4'),xlabel('pulsatia'),ylabel('Ampl 4');
end;                %if
semilogx(w,arg4),grid,title('c.f.p.4'),xlabel('pulsatia'),ylabel('
Arg 4');
pause
end                   %ifk11=4

if k11==5
  if (real(p5)<=0.00001),disp('Sistem stabil')
     else, disp(''),disp('    Pentru H5(s) exista poli
nenegativi')
     disp('                  ==> sistemul este instabil !!!');
end             %if
 [re5,im5]=nyquist(num5,den5,w);
 [mod5,arg5]=bode(num5,den5,w);
 modd5=20*log10(mod5);
```

```
semilogx(w,modd5,'w'),grid,title('c.a.p.5'),xlabel('pulsatia'),yla
bel('Ampl 5');
pause
if (real(p5)<0),loglog(w,modd5),grid;
  title('c.a.p.5'),xlabel('pulsatia'),ylabel('Ampl 5');
end;                %if
semilogx(w,arg5),grid,title('c.f.p.5'),xlabel('pulsatia'),ylabel('
Arg 5');
pause
end                   %ifk11=5

if k11==6
  if (real(p6)<=0.00001),disp('Sistem stabil')
     else, disp(''),disp('    Pentru H6(s) exista poli
nenegativi')
     disp('                   ==> sistemul este instabil !!!');
end              %if
 [re6,im6]=nyquist(num6,den6,w);
 [mod6,arg6]=bode(num6,den6,w);
modd6=20*log10(mod6);
semilogx(w,modd6,'w'),grid,title('c.a.p.6'),xlabel('pulsatia'),yla
bel('Ampl 6');
pause
if (real(p6)<0),loglog(w,modd6),grid;
  title('c.a.p.6'),xlabel('pulsatia'),ylabel('Ampl 6');
end;                %if
semilogx(w,arg6),grid,title('c.f.p.6'),xlabel('pulsatia'),ylabel('
Arg 6');
pause
end                   %ifk11=6

if k11==7
  if (real(p7)<=0.00001),disp('Sistem stabil')
     else, disp(''),disp('    Pentru H7(s) exista poli
nenegativi')
     disp('                   ==> sistemul este instabil !!!');
end              %if
 [re7,im7]=nyquist(num7,den7,w);
 [mod7,arg7]=bode(num7,den7,w);
modd7=20*log10(mod7);
semilogx(w,modd7,'w'),grid,title('c.a.p.7'),xlabel('pulsatia'),yla
bel('Ampl 7');
pause
if (real(p7)<0),loglog(w,modd7),grid;
  title('c.a.p.7'),xlabel('pulsatia'),ylabel('Ampl 7');
end;                %if
semilogx(w,arg7),grid,title('c.f.p.7'),xlabel('pulsatia'),ylabel('
Arg 7');
pause
end                  %ifk11=7
```

```
if k11==8
   if (real(p8)<=0.00001),disp('Sistem stabil')
        else, disp(''),disp('    Pentru H8(s) exista poli
nenegativi')
        disp('                   ==> sistemul este instabil !!!');
    end                %if
   [re8,im8]=nyquist(num8,den8,w);
   [mod8,arg8]=bode(num8,den8,w);
   modd8=20*log10(mod8);
   semilogx(w,modd8,'w'),grid,title('c.a.p.8'),xlabel('pulsatia'),
ylabel('Ampl 8');
    pause
    if (real(p8)<0),loglog(w,modd8),grid;
       title('c.a.p.8'),xlabel('pulsatia'),ylabel('Ampl 8');
    end;                  %if
    semilogx(w,arg8),grid,title('c.f.p.8'),xlabel('pulsatia'),
ylabel('Arg 8');
    pause
end;                    %ifk11=8

if k11==9
  if (real(p9)<=0.00001),disp('Sistem stabil')
     else, disp(''),disp('    Pentru H9(s) exista poli
nenegativi')
     disp('                   ==> sistemul este instabil !!!');
end             %if
 [re9,im9]=nyquist(num9,den9,w);
 [mod9,arg9]=bode(num9,den9,w);
modd9=20*log10(mod9);
semilogx(w,modd9,'w'),grid,title('c.a.p.9'),xlabel('pulsatia'),yla
bel('Ampl 9');
pause
if (real(p9)<0),loglog(w,modd9),grid;
  title('c.a.p.9'),xlabel('pulsatia'),ylabel('Ampl 9');
end;               %if
semilogx(w,arg9),grid,title('c.f.p.9'),xlabel('pulsatia'),ylabel('
Arg 9');
pause
end                  %ifk11=10

if k11==10
  if (real(p10)<=0.00001),disp('Sistem stabil')
     else, disp(''),disp('    Pentru H10(s) exista poli
nenegativi')
     disp('                   ==> sistemul este instabil !!!');
end             %if
 [re10,im10]=nyquist(num10,den10,w);
 [mod10,arg10]=bode(num10,den10,w);
modd10=20*log10(mod10);
semilogx(w,modd10,'w'),grid,title('c.a.p.10'),xlabel('pulsatia'),y
label('Ampl 10');
```

```
pause
if (real(p10)<0),loglog(w,modd10),grid;
  title('c.a.p.10'),xlabel('pulsatia'),ylabel('Ampl 10');
end;               %if
semilogx(w,arg10),grid,title('c.f.p.10'),xlabel('pulsatia'),ylabel
('Arg 10');
pause
end                  %ifk11=10

if k11==11
     hold off
w=logspace(1,2.5,100);

t100=0;
while t100==0
    k100=menu('Caracteristici_frecventa','Pe
rand','H1','H2','H3','H4','H5','H6','H7','H8','H9','H10','Suprapus
e','Revenire');
%Hi=functii de transfer=zi/u
  if k100==1
    hold off
    [mod1,arg1]=bode(num1,den1,w);
    semilogx(w,mod1,'r'),grid,title('c.a.p.1'),xlabel('pulsatia'),
ylabel('Ampl');pause
    hold on
    [mod2,arg2]=bode(num2,den2,w);
    semilogx(w,mod2,'b'),grid,title('c.a.p.1,2'),xlabel
('pulsatia'), ylabel('Ampl');pause
    [mod3,arg3]=bode(num3,den3,w);
    semilogx(w,mod3,'y'),grid,title('c.a.p.1..3'),xlabel
('pulsatia'), ylabel('Ampl');pause
    [mod4,arg4]=bode(num4,den4,w);
    semilogx(w,mod4,'m'),grid,title('c.a.p.1..4'),xlabel
('pulsatia'), ylabel('Ampl');pause
    [mod5,arg5]=bode(num5,den5,w);
    semilogx(w,mod5,'c'),grid,title('c.a.p.1..5'),xlabel
('pulsatia'), ylabel('Ampl');pause
    [mod6,arg6]=bode(num6,den6,w);
    semilogx(w,mod6,'w'),grid,title('c.a.p.1..6'),xlabel
('pulsatia'), ylabel('Ampl');pause
    [mod7,arg7]=bode(num7,den7,w);
    semilogx(w,mod7,'r'),grid,title('c.a.p.1..7'),xlabel
('pulsatia'), ylabel('Ampl');pause
    [mod8,arg8]=bode(num8,den8,w);
    semilogx(w,mod8,'c'),grid,title('c.a.p.1..8'),xlabel
('pulsatia'), ylabel('Ampl');pause
    [mod9,arg9]=bode(num9,den9,w);
    semilogx(w,mod9,'w'),grid,title('c.a.p.1..9'),xlabel
('pulsatia'), ylabel('Ampl');pause
    [mod10,arg10]=bode(num10,den10,w);
    semilogx(w,mod10,'r'),grid,title('c.a.p.1..10'),xlabel
('pulsatia'), ylabel('Ampl');pause
    hold off
  end     %ifk100=1

  if k100==2
    [mod1,arg1]=bode(num1,den1,w);
    semilogx(w,mod1,'r'),grid,title('c.a.p.1'),xlabel('pulsatia'),
ylabel('Ampl');pause
  end     %ifk100=2

  if k100==3
    [mod2,arg2]=bode(num2,den2,w);
    semilogx(w,mod2,'b'),grid,title('c.a.p.2'),xlabel('pulsatia'),
ylabel('Ampl');pause
  end     %ifk100=3

  if k100==4
    [mod3,arg3]=bode(num3,den3,w);
    semilogx(w,mod3,'y'),grid,title('c.a.p.3'),xlabel('pulsatia'),
ylabel('Ampl');pause
  end      %ifk100=4

  if k100==5
    [mod4,arg4]=bode(num4,den4,w);
    semilogx(w,mod4,'m'),grid,title('c.a.p.4'),xlabel('pulsatia'),
ylabel('Ampl');pause
  end      %ifk100=5

  if k100==6
    [mod5,arg5]=bode(num5,den5,w);
    semilogx(w,mod5,'c'),grid,title('c.a.p.5'),xlabel('pulsatia'),
ylabel('Ampl');pause
  end      %ifk100=6

  if k100==7
    [mod6,arg6]=bode(num6,den6,w);
    semilogx(w,mod6,'w'),grid,title('c.a.p.6'),xlabel('pulsatia'),
ylabel('Ampl');pause
  end      %ifk100=7

  if k100==8
    [mod7,arg7]=bode(num7,den7,w);
    semilogx(w,mod7,'r'),grid,title('c.a.p.7'),xlabel('pulsatia'),
ylabel('Ampl');pause
  end      %ifk100=8

  if k100==9
    [mod8,arg8]=bode(num8,den8,w);
    semilogx(w,mod8,'c'),grid,title('c.a.p.8'),xlabel('pulsatia'),
ylabel('Ampl');pause
```

```
  end      %ifk100=9

  if k100==10
    [mod9,arg9]=bode(num9,den9,w);
    semilogx(w,mod9,'w'),grid,title('c.a.p.9'),xlabel('pulsatia'),
ylabel('Ampl');pause
  end      %ifk100=10

  if k100==11
    [mod10,arg10]=bode(num10,den10,w);
    semilogx(w,mod10,'r'),grid,title('c.a.p.10'),xlabel
('pulsatia'), ylabel('Ampl');pause
  end      %ifk100=11

 if k100==12
    hold off
    [mod1,arg1]=bode(num1,den1,w);
    semilogx(w,mod1,'r'),grid,title('c.a.p.1'),xlabel('pulsatia'),
ylabel('Ampl');
    hold on
    [mod2,arg2]=bode(num2,den2,w);
    semilogx(w,mod2,'b'),grid,title('c.a.p.1,2'),xlabel
('pulsatia'), ylabel('Ampl');
    [mod3,arg3]=bode(num3,den3,w);
    semilogx(w,mod3,'y'),grid,title('c.a.p.1..3'),xlabel
('pulsatia'),ylabel('Ampl');
    [mod4,arg4]=bode(num4,den4,w);
    semilogx(w,mod4,'m'),grid,title('c.a.p.1..4'),xlabel
('pulsatia'),ylabel('Ampl');
    [mod5,arg5]=bode(num5,den5,w);
    semilogx(w,mod5,'c'),grid,title('c.a.p.1..5'),xlabel
('pulsatia'),ylabel('Ampl');
    [mod6,arg6]=bode(num6,den6,w);
    semilogx(w,mod6,'w'),grid,title('c.a.p.1..6'),xlabel
('pulsatia'),ylabel('Ampl');
    [mod7,arg7]=bode(num7,den7,w);
    semilogx(w,mod7,'r'),grid,title('c.a.p.1..7'),xlabel
('pulsatia'),ylabel('Ampl');
    [mod8,arg8]=bode(num8,den8,w);
    semilogx(w,mod8,'c'),grid,title('c.a.p.1..8'),xlabel
('pulsatia'),ylabel('Ampl');
    [mod9,arg9]=bode(num9,den9,w);
    semilogx(w,mod9,'w'),grid,title('c.a.p.1..9'),xlabel
('pulsatia'),ylabel('Ampl');
    [mod10,arg10]=bode(num10,den10,w);
    semilogx(w,mod10,'r'),grid,title('c.a.p.1..10'),xlabel
('pulsatia'),ylabel('Ampl');pause
    hold off
  end     %ifk100=9

  if k100==13
```

```
     t100=1;
  end     %ifk100=10
 end     %while_t100
end                 %ifk11=8

if k11==12
     hold off
     w=logspace(1,2.5,100);
     [mod1,arg1]=bode(num1,den1,w);
     modd1=20*log10(mod1);
     semilogx(w,modd1,'w'),grid,title('c.a.p.1'),xlabel
('pulsatia'),ylabel('Ampl');
     hold on
     [mod2,arg2]=bode(num2,den2,w);
     modd2=20*log10(mod2);
     semilogx(w,modd2,'w'),grid,title('c.a.p.2'),xlabel
('pulsatia'),ylabel('Ampl');
     [mod3,arg3]=bode(num3,den3,w);
     modd3=20*log10(mod3);
     semilogx(w,modd3,'w'),grid,title('c.a.p.3'),xlabel
('pulsatia'),ylabel('Ampl');
     [mod4,arg4]=bode(num4,den4,w);
     modd4=20*log10(mod4);
     semilogx(w,modd4,'w'),grid,title('c.a.p.4'),xlabel
('pulsatia'),ylabel('Ampl');
     [mod5,arg5]=bode(num5,den5,w);
     modd5=20*log10(mod5);
     semilogx(w,modd5,'w'),grid,title('c.a.p.5'),xlabel
('pulsatia'),ylabel('Ampl');
     [mod6,arg6]=bode(num6,den6,w);
     modd6=20*log10(mod6);
     semilogx(w,modd6,'w'),grid,title('c.a.p.6'),xlabel
('pulsatia'),ylabel('Ampl');
     [mod7,arg7]=bode(num7,den7,w);
     modd7=20*log10(mod7);
     semilogx(w,modd7,'w'),grid,title('c.a.p.7'),xlabel
('pulsatia'),ylabel('Ampl');
    [mod8,arg8]=bode(num8,den8,w);
     modd8=20*log10(mod8);
     semilogx(w,modd8,'w'),grid,title('c.a.p.8'),xlabel
('pulsatia'),ylabel('Ampl');
    [mod9,arg9]=bode(num9,den9,w);
     modd9=20*log10(mod9);
     semilogx(w,modd9,'w'),grid,title('c.a.p.9'),xlabel
('pulsatia'),ylabel('Ampl');
    [mod10,arg10]=bode(num10,den10,w);
     modd10=20*log10(mod10);
     semilogx(w,modd10,'w'),grid,title('c.a.p.10'),xlabel
'pulsatia'),ylabel('Ampl');
     grid
    pause
    hold off
end                 %ifk11=12

if k11==13;
  t11=1;
end;                %ifk11=13
end;                %while_t11

%Generare semnal de intrare: semnal sinusoidal
%n=input('Dati lungimea secventei  n=');
%F0=input('Dati amplitudinea fortei F0=');
tfin=0;
while tfin==0
  t2=0;
  while t2==0
    k2=menu('SEMNAL
INTRARE','SINUSOIDAL','IMPULS','TREAPTA','CONTINUE');
    if k2==1
     k6=menu('init','de la tastatura','implicite');
     if k6==1
         n=input('Dati lungimea secventei  n=');
         x0=input('Dati amplitudinea miscarii x0=');
         F0=k*x0;
         k7=menu('Dati','pulsatia','frecventa');
           if k7==1
             omega=input('Dati pulsatia:');
             niu=omega/(2*pi);         %frecventa
           end;                          %ifk7=1
           if k7==2
             niu=input('Dati frecventa, in Hz:  ');
           end;                          %ifk7=2
     end;                %ifk6=1
     if k6==2
         n=3;
         x0=0.01;
         F0=k*0.01;
         niu=10;
     end;                %ifk6=2
     T=1/niu;            %calculul perioadei oscilatiilor
     i=0:0.01:n;
     u=x0*sin(2*pi*i./T);
     p=u';
     r=u';
     plot(0:0.01:n,u),grid,title('Semnal intrare  xf - sinus');
     xlabel('t'),ylabel('x0');
     pause;
     [y,t]=lsim(Am,Bm,Cm,Dm,[r,p],0:0.01:n);
     [y1,t]=lsim(Am1,Bm1,Cm1,Dm1,[r,p],0:0.01:n);
     [y2,t]=lsim(Am2,Bm2,Cm2,Dm2,[r,p],0:0.01:n);
     [y3,t]=lsim(Am3,Bm3,Cm3,Dm3,[r,p],0:0.01:n);
     [y4,t]=lsim(Am4,Bm4,Cm4,Dm4,[r,p],0:0.01:n);
```

```
 [y5,t]=lsim(Am5,Bm5,Cm5,Dm5,[r,p],0:0.01:n);
 [y6,t]=lsim(Am6,Bm6,Cm6,Dm6,[r,p],0:0.01:n);
 [y7,t]=lsim(Am7,Bm7,Cm7,Dm7,[r,p],0:0.01:n);
 [y8,t]=lsim(Am8,Bm8,Cm8,Dm8,[r,p],0:0.01:n);
 [y9,t]=lsim(Am9,Bm9,Cm9,Dm9,[r,p],0:0.01:n);
 [y10,t]=lsim(Am10,Bm10,Cm10,Dm10,[r,p],0:0.01:n);
end;                    %ifk2=1

if k2==2
 k6=menu('VALORI INPUT','de la tastatura','implicite');
 if k6==1
     n=input('Dati lungimea secventei  n=');
     F0=input('Dati amplitudinea fortei F0=');
     b=input('Dati lungimea impulsului b<n,   b=');
 end;                %ifk6=1
 if k6==2
     n=10;
     F0=200;
     b=1;
 end;                %ifk6=2
 u=zeros(1,n);
    for i=1:0.01:b
      u(i)=F0;
      p=u';
    end;                %for
 plot(1:n,u),grid,title('FORTA PERTURBATOARE  - impuls');
 xlabel('t'),ylabel('F');
 pause;

 [y,t]=lsim(Am,Bm,Cm,Dm,[r,p],0:0.01:n);
 [y1,t]=lsim(Am1,Bm1,Cm1,Dm1,[r,p],0:0.01:n);
 [y2,t]=lsim(Am2,Bm2,Cm2,Dm2,[r,p],0:0.01:n);
 [y3,t]=lsim(Am3,Bm3,Cm3,Dm3,[r,p],0:0.01:n);
 [y4,t]=lsim(Am4,Bm4,Cm4,Dm4,[r,p],0:0.01:n);
 [y5,t]=lsim(Am5,Bm5,Cm5,Dm5,[r,p],0:0.01:n);
 [y6,t]=lsim(Am6,Bm6,Cm6,Dm6,[r,p],0:0.01:n);
 [y7,t]=lsim(Am7,Bm7,Cm7,Dm7,[r,p],0:0.01:n);
 [y8,t]=lsim(Am8,Bm8,Cm8,Dm8,[r,p],0:0.01:n);
 [y9,t]=lsim(Am9,Bm9,Cm9,Dm9,[r,p],0:0.01:n);
 [y10,t]=lsim(Am10,Bm10,Cm10,Dm10,[r,p],0:0.01:n);
end;                    %ifk2=2

if k2==3
 k6=menu('init','de la tastatura','implicite');
 if k6==1
     n=input('Dati lungimea secventei  n=');
     F0=input('Dati amplitudinea fortei F0=');
 end;                %ifk6=1
 if k6==2
     n=10;
     F0=1;
     end;                %ifk6=2
     for i=0:n
          u(i)=F0;
          p=u';
     end;                %for
     plot(0:n,u),grid,title('FORTA PERTURBATOARE - treapta');
     xlabel('t'),ylabel('F');
     pause;
     [y,t]=lsim(Am,Bm,Cm,Dm,[r,p],0:0.01:n);
     [y1,t]=lsim(Am1,Bm1,Cm1,Dm1,[r,p],0:0.01:n);
     [y2,t]=lsim(Am2,Bm2,Cm2,Dm2,[r,p],0:0.01:n);
     [y3,t]=lsim(Am3,Bm3,Cm3,Dm3,[r,p],0:0.01:n);
     [y4,t]=lsim(Am4,Bm4,Cm4,Dm4,[r,p],0:0.01:n);
     [y5,t]=lsim(Am5,Bm5,Cm5,Dm5,[r,p],0:0.01:n);
     [y6,t]=lsim(Am6,Bm6,Cm6,Dm6,[r,p],0:0.01:n);
     [y7,t]=lsim(Am7,Bm7,Cm7,Dm7,[r,p],0:0.01:n);
     [y8,t]=lsim(Am8,Bm8,Cm8,Dm8,[r,p],0:0.01:n);
     [y9,t]=lsim(Am9,Bm9,Cm9,Dm9,[r,p],0:0.01:n);
     [y10,t]=lsim(Am10,Bm10,Cm10,Dm10,[r,p],0:0.01:n);
    end;                        %ifk2=3

    if k2==4
     t2=1;
    end;                        %ifk2=4
end;                            %while_t2

%afisarea grafica a raspunsului
t3=0;
while t3==0
    k3=menu('AFISARE','un
parametru','all','all1','all2','all3','all4','REVENIRE','EXIT');
    if k3==1
     t4=0;
     while t4==0
       k4=menu('PARAMETRI','z1','z2','z3','z4','z5','z6', 'z7',
'z8','z9','z10','Quit');
         if k4==1
          plot(y1);grid;xlabel('t');ylabel('z1');
          pause;
         end;            %ifk4=1
         if k4==2
          plot(y2),grid,xlabel('t'),ylabel('z2');
          pause;
         end;            %ifk4=2
         if k4==3
          plot(y3),grid,xlabel('t'),ylabel('z3');
          pause;
         end;            %ifk4=3
         if k4==4
          plot(y4);grid;xlabel('t');ylabel('z4');
          pause;
         end;               %ifk4=4
         if k4==5
          plot(y5);grid;xlabel('t');ylabel('z5');
          pause;
         end;               %ifk4=5
         if k4==6
          plot(y6);grid;xlabel('t');ylabel('z6');
          pause;
         end;               %ifk4=6
         if k4==7
          plot(y7);grid;xlabel('t');ylabel('z7');
          pause;
         end;               %ifk4=7
         if k4==8
          plot(y8);grid;xlabel('t');ylabel('z8');
          pause;
         end;               %ifk4=8
         if k4==9
          plot(y9);grid;xlabel('t');ylabel('z9');
          pause;
         end;               %ifk4=9
         if k4==10
          plot(y10);grid;xlabel('t');ylabel('z10');
          pause;
         end;               %ifk4=10

         if k4==11
          t4=1;
         end;               %ifk4=11
     end;                   %while_t4
    end;                       %ifk3=1
    if k3==2
        %hold on
     plot(y);grid;xlabel('t');ylabel('y');
     pause;
    end;                       %ifk3=2

    if k3==3
     subplot(111),plot(y1,'w'),grid,xlabel('t'),ylabel('z1'),
pause;
     plot(y2,'w'),grid,xlabel('t'),ylabel('z2');pause;
     plot(y3,'w'),grid,xlabel('t'),ylabel('z3');pause;
     plot(y4,'w'),grid,xlabel('t'),ylabel('z4');pause;
     plot(y5,'w'),grid,xlabel('t'),ylabel('z5');pause;
     plot(y6,'w'),grid,xlabel('t'),ylabel('z6');pause;
     plot(y7,'w'),grid,xlabel('t'),ylabel('z7');pause;
     plot(y8,'w'),grid,xlabel('t'),ylabel('z8');pause;
     plot(y9,'w'),grid,xlabel('t'),ylabel('z9');pause;
     plot(y10,'w'),grid,xlabel('t'),ylabel('z10');pause;
        end;                    %ifk3=3
    if k3==4
       subplot(221),plot(y1,'w'),grid;xlabel('t');ylabel('z1');
pause
        subplot(222),plot(y2,'w'),grid;xlabel('t');ylabel('z2');
pause
        subplot(223),plot(y3,'w'),grid;xlabel('t');ylabel('z3');
pause
        subplot(224),plot(y4,'w'),grid;xlabel('t');ylabel('z4');
pause
        clg;
        subplot(221),plot(y5,'w'),grid;xlabel('t');ylabel('z5');
pause
        subplot(222),plot(y6,'w'),grid;xlabel('t');ylabel('z6');
pause
        subplot(223),plot(y7,'w'),grid;xlabel('t');ylabel('z7');
pause
        subplot(224),plot(y8,'w'),grid;xlabel('t');ylabel('z8');
pause
     clg;
        subplot(221),plot(y9,'w'),grid;xlabel('t');ylabel('z9');
pause
        subplot(222),plot(y10,'w'),grid;xlabel('t');ylabel('z10');
pause
     pause;
     subplot(111);
     clg
end;                    %ifk3=4

    if k3==5
        subplot(1,1,1);
        hold off;
        plot(u,'w');
        hold on
        plot(y);grid;xlabel('t');ylabel('y');
        pause;
        hold off;
    end;                    %ifk3=5

    if k3==6
        hold off
        hold on;
        subplot(111)
        plot(u,'w');
        pause
        hold on
        plot(y1,'m'),grid,xlabel('t'),ylabel('z1'),pause;
        plot(y2,'r'),grid,xlabel('t'),ylabel('z1,z2');pause;
        plot(y3,'r'),grid,xlabel('t'),ylabel('z1..z3');pause;
        plot(y4,'c'),grid,xlabel('t'),ylabel('z1..z4');pause;
        plot(y5,'y'),grid,xlabel('t'),ylabel('z1..z5');pause;
        plot(y6,'b'),grid,xlabel('t'),ylabel('z1..z6');pause;
        plot(y7,'w'),grid,xlabel('t'),ylabel('z1..z7');pause;
        plot(y8,'y'),grid,xlabel('t'),ylabel('z1..z8');pause;
        plot(y9,'b'),grid,xlabel('t'),ylabel('z1..z9');pause;
        plot(y10,'w'),grid,xlabel('t'),ylabel('z1..z10');pause;
        hold off
    end;                    %ifk3=6

    if k3==7
     t3=1;
    end   %ifk3=7

    if k3==8
     t3=1;
     tfin=1;
    end;                    %ifk3=8
  end;                        %while_t3
end;   %while_tfin

cla;
disp(''),disp(''),disp(''),disp(''),disp(''),disp(''),disp(''),
disp('');
disp(''),disp('');
disp('               VA MULTUMIM CA ATI FOLOSIT ACEST PROGRAM');
disp(''),disp(''),disp(''),disp(''),disp(''),disp('');
disp('  prof.coord.:
doctorand:');
disp('  prof.univ.dr.ing.Liviu BRINDEU
ing.Cornel POPA');
disp(''),disp(''),disp(''),disp(''),disp(''),disp('');
end                         %program
```

ANEXA 7.1

Codul sursă al programului de simulare, scris folosind mediul de programare Mathlab, este redat în continuare. Numele programului este *kcasez.m*.

```
%Program pentru simularea functionarii unui model mecanic
%al corpului omenesc
%stergere ecran in mod text
cla
%stergere ecran in mod grafic
clg
%stergerea variabilelor existente in memorie
clear
disp('   M O D E L    M E C A N I C');
disp('   ~~~~~~~~~    ~~~~~~~~~~~~~');
format compact;
i=sqrt(-1);
%introducerea datelor initiale:
t1=0;
while t1==0
    k1=menu('INITIALIZARI','de         la         TASTATURA','date
implicite','continue');
    if k1==1
     disp('Introduceti datele initiale ale problemei !');
     k12=input('k1=');
     k27=input('k2=');
     k23=input('k3=');
     k34=input('k4=');
     k45=input('k5=');
     k56=input('k6=');
     k67=input('k7=');
     k8=input('k8=');

        k21=menu('TIPUL      MODELULUI','cu      amortizare','fara
amortizare');
        if k21==1

     c12=input('c1=');
     c27=input('c2=');
     c23=input('c3=');
     c34=input('c4=');
     c45=input('c5=');
     c56=input('c6=');
     c67=input('c7=');
     c8=input('c8=');

        end;               %ifk21=1
        if k21==2
           c12=0;c23=0;c34=0;c45=0;c56=0;c67=0;c27=0;c8=0;
```

```
        end;               %ifk21=2

     t6=0;
     while t6==0
     k6=menu('masa corpului','toate masele componente','doar masa
corpului');
     if k6==1
         m1=input('m1=');
         m2=input('m2=');
         m3=input('m3=');
         m4=input('m4=');
         m5=input('m5=');
         m6=input('m6=');
         m7=input('m7=');
%t1=1;
         t6=1;
     end;                %ifk6=1
     if k6==2
         m=input('m=');
         m1=6.82*m/100;
         m2=8.52*m/100;
         m3=40.91*m/100;
         m4=1.71*m/100;
         m5=0.57*m/100;
         m6=7.38*m/100;
         m7=34.09*m/100;
         t6=1;
      end;              %ifk6=2
     end;          %while_t6
    end;                %ifk1=1
    if k1==2;

     m1=5.44;
     m2=6.82;
     m3=32.72;
     m4=1.36;
     m5=0.46;
     m6=5.91;
     m7=27.22;

 k20=menu(' ','cu amortizare','fara amortizare');
   if k20==1
     k12=48953.24;
     k23=25526.83;
     k34=715.88;
     k45=715.88;
     k56=1118.79;
     k67=1118.79;
     k27=40634.65;
     k8=22845.87;
     c12=103.36;
     c23=90.47;
     c34=11.03;
     c45=11.03;
     c56=28.73;
     c67=28.73;
     c27=107.14;
     end;               %ifk20=1

   if k20==2
     k12=48463.71;
     k23=25429.87;
     k34=671.13;
     k45=671.13;
     k56=1048.86;
     k67=1048.86;
     k27=44709.86;
     k8=23147.22;
     c12=0;c23=0;c34=0;c45=0;c56=0;c67=0;c27=0;c8=0;
        end;               %ifk20=2
    end;                    %ifk1=2

   if k1==3
     t1=1;
    end;                    %ifk1=3
end;                        %while t1

%definirea numaratorilor si numitorilor functiilor de transfer
%corespunzatoare blocurilor componente
n1=1;
d1=1;
n2=k27;
d2=1;
n3=c27;
d3=1;
n4=k56;
d4=1;
n5=c56;
d5=1;
n6=1;
d6=[m7 0];
n7=1;
d7=[1 0];
n8=k56;
d8=1;
n9=c56;
d9=1;
n10=1;
d10=[m6 0];
n11=2;
d11=1;
```

```
n12=2;
d12=1;
n13=1;
d13=[1 0];
n14=k56;
d14=1;
n15=c56;
d15=1;
n16=k34;
d16=1;
n17=c34;
d17=1;
n18=1;
d18=[m5 0];
n19=1;
d19=[1 0];
n20=k34;
d20=1;
n21=c34;
d21=1;
n22=1;
d22=[m4 0];
n23=2;
d23=1;
n24=1;
d24=[1 0];
n25=2;
d25=1;
n26=k34;
d26=1;
n27=c34;
d27=1;
n28=k23;
d28=1;
n29=c23;
d29=1;
n30=1;
d30=[m3 0];
n31=1;
d31=[1 0];
n32=k23;
d32=1;
n33=c23;
d33=1;
n34=k12;
d34=1;
n35=c12;
d35=1;
n36=1;
d36=[m2 0];
n37=c12;
```

```
d37=1;
n38=1;
d38=[1 0];
n39=k27;
d39=1;
n40=k12;
d40=1;
n41=1;
d41=[m1 0];
n42=c12;
d42=1;
n43=1;
d43=[1 0];
n44=k8;
d44=1;

%definirea numarului de blocuri
nblocks=44;
blkbuild;
%specificarea interconexiunilor dintre blocuri
q=[1 0 0 0 0 0 0
   2 -7 38 0 0 0 0
   3 -6 36 0 0 0 0
   4 -7 13 0 0 0 0
   5 -6 10 0 0 0 0
   6 2 3 4 5 44 0
   7 6 0 0 0 0 0
   8 7 -12 19 0 0 0
   9 6 -11 18 0 0 0
   10 8 9 0 0 0 0
   11 10 0 0 0 0 0
   12 13 0 0 0 0 0
   13 10 0 0 0 0 0
   14 13 -19 0 0 0 0
   15 10 -18 0 0 0 0
   16 -19 24 0 0 0 0
   17 -18 22 0 0 0 0
   18 14 15 16 17 0 0
   19 18 0 0 0 0 0
   20 19 -25 31 0 0 0
   21 18 -23 30 0 0 0
   22 20 21 0 0 0 0
   23 22 0 0 0 0 0
   24 22 0 0 0 0 0
   25 24 0 0 0 0 0
   26 24 -31 0 0 0 0
   27 22 -30 0 0 0 0
   28 -31 38 0 0 0 0
   29 -30 36 0 0 0 0
   30 26 27 28 29 0 0
   31 30 0 0 0 0 0
   32 31 -38 0 0 0 0
   33 30 -36 0 0 0 0
   34 -38 43 0 0 0 0
   35 -36 41 0 0 0 0
   36 32 33 34 35 37 39
   37 6 -36 0 0 0 0
   38 36 0 0 0 0 0
   39 7 -38 0 0 0 0
   40 38 -43 0 0 0 0
   41 40 42 0 0 0 0
   42 36 -41 0 0 0 0
   43 41 0 0 0 0 0
   44 1 -7 0 0 0 0];

%specificarea intrarii si iesirilor
iu=[1];
iy=[43 38 31 24 19 13 7];
%interconectarea propriu-zisa
[A B C D]=connect(a,b,c,d,q,iu,iy);
%realizarea unui model minimal
[Am Bm Cm Dm]=minreal(A,B,C,D);
%prin MM-ISI avand matricile Am,Bm,Cm,Dm

iy1=43;
[A1 B1 C1 D1]=connect(a,b,c,d,q,iu,iy1);
[Am1 Bm1 Cm1 Dm1]=minreal(A1,B1,C1,D1);

iy2=38;
[A2 B2 C2 D2]=connect(a,b,c,d,q,iu,iy2);
[Am2 Bm2 Cm2 Dm2]=minreal(A2,B2,C2,D2);

iy3=31;
[A3 B3 C3 D3]=connect(a,b,c,d,q,iu,iy3);
[Am3 Bm3 Cm3 Dm3]=minreal(A3,B3,C3,D3);

iy4=24;
[A4 B4 C4 D4]=connect(a,b,c,d,q,iu,iy4);
[Am4 Bm4 Cm4 Dm4]=minreal(A4,B4,C4,D4);

iy5=19;
[A5 B5 C5 D5]=connect(a,b,c,d,q,iu,iy5);
[Am5 Bm5 Cm5 Dm5]=minreal(A5,B5,C5,D5);

iy6=13;
[A6 B6 C6 D6]=connect(a,b,c,d,q,iu,iy6);
[Am6 Bm6 Cm6 Dm6]=minreal(A6,B6,C6,D6);

iy7=7;
[A7 B7 C7 D7]=connect(a,b,c,d,q,iu,iy7);
[Am7 Bm7 Cm7 Dm7]=minreal(A7,B7,C7,D7);
```

```
[num1 den1]=ss2tf(Am1,Bm1,Cm1,Dm1,iu);
%pause;
 [z1,p1,k]=tf2zp(num1,den1);

[num2 den2]=ss2tf(Am2,Bm2,Cm2,Dm2,iu);
%pause;

 [z2,p2,k]=tf2zp(num2,den2);

[num3 den3]=ss2tf(Am3,Bm3,Cm3,Dm3,iu);
%pause;
[z3,p3,k]=tf2zp(num3,den3);

[num4 den4]=ss2tf(Am4,Bm4,Cm4,Dm4,iu);
%pause;
[z4,p4,k]=tf2zp(num4,den4);

[num5 den5]=ss2tf(Am5,Bm5,Cm5,Dm5,iu);
%pause;
[z5,p5,k]=tf2zp(num5,den5);

[num6 den6]=ss2tf(Am6,Bm6,Cm6,Dm6,iu);
%pause;
[z6,p6,k]=tf2zp(num6,den6);
%pause

[num7 den7]=ss2tf(Am7,Bm7,Cm7,Dm7,iu);
%pause;
[z7,p7,k]=tf2zp(num7,den7);
%pause

Am1;
%pause
Bm1;
%pause
Cm1;
%pause
Dm1;
%pause

w=logspace(-1,3,100);

t11=0;
while t11==0

k11=menu('CARACTERISTICI_FRECVENTA','H1','H2','H3','H4','H5','H6',
'H7','PE RAND','SUPRAPUSE','CONTINUE');
%Hi=functii de transfer=zi/u
    if k11==1
      if (real(p1)<=0.00001),disp('Sistem stabil')
```

```
        else,  disp(''),disp('         Pentru  H1(s)  exista  poli
nenegativi')
        disp('                   ==> sistemul este instabil !!!');
    end             %if
    [re1,im1]=nyquist(num1,den1,w);
    [mod1,arg1]=bode(num1,den1,w);
    modd1=20*log10(mod1);
    semilogx(w,modd1,'w'),grid,title('c.a.p.1'),xlabel
('pulsatia'), ylabel('Ampl 1');
    pause
    if (real(p1)<0),loglog(w,modd1),grid;
       title('c.a.p.1'),xlabel('pulsatia'),ylabel('Ampl 1');
    end;                %if
    semilogx(w,arg1),grid,title('c.f.p.1'),xlabel('pulsatia'),
ylabel('Arg 1');
    pause
end;                    %ifk11=1

if k11==2
  if (real(p2)<=0.00001),disp('Sistem stabil')
     else,  disp(''),disp('         Pentru  H2(s)  exista  poli
nenegativi')
     disp('                  ==> sistemul este instabil !!!');
end             %if
[re2,im2]=nyquist(num2,den2,w);
[mod2,arg2]=bode(num2,den2,w);
modd2=20*log10(mod2);
semilogx(w,modd2,'w'),grid,title('c.a.p.2'),xlabel('pulsatia'),yla
bel('Ampl 2');
pause
if (real(p2)<0),loglog(w,modd2),grid;
  title('c.a.p.2'),xlabel('pulsatia'),ylabel('Ampl 2');
end;                  %if
semilogx(w,arg2),grid,title('c.f.p.2'),xlabel('pulsatia'),ylabel('
Arg 2');
pause
end                     %ifk11=2

if k11==3
  if (real(p3)<=0.00001),disp('Sistem stabil')
     else,  disp(''),disp('         Pentru  H3(s)  exista  poli
nenegativi')
     disp('                  ==> sistemul este instabil !!!');
end             %if
 [re3,im3]=nyquist(num3,den3,w);
 [mod3,arg3]=bode(num3,den3,w);
modd3=20*log10(mod3);
semilogx(w,modd3,'w'),grid,title('c.a.p.3'),xlabel('pulsatia'),yla
bel('Ampl 3');
pause
if (real(p3)<0),loglog(w,modd3),grid;
```

```
  title('c.a.p.3'),xlabel('pulsatia'),ylabel('Ampl 3');
end;               %if
semilogx(w,arg3),grid,title('c.f.p.3'),xlabel('pulsatia'),ylabel('
Arg 3');
pause
end                  %ifk11=3

if k11==4
  if (real(p4)<=0.00001),disp('Sistem stabil')
     else,   disp(''),disp('          Pentru   H4(s)   exista   poli
nenegativi')
     disp('                   ==> sistemul este instabil !!!');
end             %if
 [re4,im4]=nyquist(num4,den4,w);
 [mod4,arg4]=bode(num4,den4,w);
modd4=20*log10(mod4);
semilogx(w,modd4,'w'),grid,title('c.a.p.4'),xlabel('pulsatia'),yla
bel('Ampl 4');
pause
if (real(p4)<0),loglog(w,modd4),grid;
  title('c.a.p.4'),xlabel('pulsatia'),ylabel('Ampl 4');
end;               %if
semilogx(w,arg4),grid,title('c.f.p.4'),xlabel('pulsatia'),ylabel('
Arg 4');
pause
end                  %ifk11=4

if k11==5
  if (real(p5)<=0.00001),disp('Sistem stabil')
     else,   disp(''),disp('          Pentru   H5(s)   exista   poli
nenegativi')
     disp('                   ==> sistemul este instabil !!!');
end             %if
 [re5,im5]=nyquist(num5,den5,w);
 [mod5,arg5]=bode(num5,den5,w);
modd5=20*log10(mod5);
semilogx(w,modd5,'w'),grid,title('c.a.p.5'),xlabel('pulsatia'),yla
bel('Ampl 5');
pause
if (real(p5)<0),loglog(w,modd5),grid;
  title('c.a.p.5'),xlabel('pulsatia'),ylabel('Ampl 5');
end;               %if
semilogx(w,arg5),grid,title('c.f.p.5'),xlabel('pulsatia'),ylabel('
Arg 5');
pause
end                  %ifk11=5

if k11==6
  if (real(p6)<=0.00001),disp('Sistem stabil')
     else,   disp(''),disp('          Pentru   H6(s)   exista   poli
nenegativi')
```

```
     disp('                   ==> sistemul este instabil !!!');
end             %if
 [re6,im6]=nyquist(num6,den6,w);
 [mod6,arg6]=bode(num6,den6,w);
modd6=20*log10(mod6);
semilogx(w,modd6,'w'),grid,title('c.a.p.6'),xlabel('pulsatia'),yla
bel('Ampl 6');
pause
if (real(p6)<0),loglog(w,modd6),grid;
  title('c.a.p.6'),xlabel('pulsatia'),ylabel('Ampl 6');
end;               %if
semilogx(w,arg6),grid,title('c.f.p.6'),xlabel('pulsatia'),ylabel('
Arg 6');
pause
end                  %ifk11=6

if k11==7
  if (real(p7)<=0.00001),disp('Sistem stabil')
     else,  disp(''),disp('          Pentru   H7(s)   exista   poli
nenegativi')
     disp('                   ==> sistemul este instabil !!!');
end             %if
 [re7,im7]=nyquist(num7,den7,w);
 [mod7,arg7]=bode(num7,den7,w);
modd7=20*log10(mod7);
semilogx(w,modd7,'w'),grid,title('c.a.p.7'),xlabel('pulsatia'),yla
bel('Ampl 7');
pause
if (real(p7)<0),loglog(w,modd7),grid;
  title('c.a.p.7'),xlabel('pulsatia'),ylabel('Ampl 7');
end;               %if
semilogx(w,arg7),grid,title('c.f.p.7'),xlabel('pulsatia'),ylabel('
Arg 7');
pause
end                 %ifk11=7

if k11==8
  hold off
  w=logspace(1,2.5,100);
  t100=0;
   while t100==0
    k100=menu('Caracteristici_frecventa','Pe  rand','H1','H2','H3',
'H4', 'H5','H6','H7','Suprapuse','Revenire');

if k100==1
    hold off
    [mod1,arg1]=bode(num1,den1,w);
    semilogx(w,mod1,'r'),grid,title('c.a.p.1'),xlabel('pulsatia'),
ylabel('Ampl');pause
    hold on
    [mod2,arg2]=bode(num2,den2,w);
    semilogx(w,mod2,'b'),grid,title('c.a.p.1,2'),xlabel
('pulsatia'), ylabel('Ampl');pause
    [mod3,arg3]=bode(num3,den3,w);
    semilogx(w,mod3,'y'),grid,title('c.a.p.1,2,3'),xlabel
('pulsatia'),ylabel('Ampl');pause
    [mod4,arg4]=bode(num4,den4,w);
    semilogx(w,mod4,'m'),grid,title('c.a.p.1,2,3,4'),xlabel
('pulsatia'),ylabel('Ampl');pause
    [mod5,arg5]=bode(num5,den5,w);
    semilogx(w,mod5,'c'),grid,title('c.a.p.1,2,3,4,5'),xlabel
('pulsatia'),ylabel('Ampl');pause
    [mod6,arg6]=bode(num6,den6,w);
    semilogx(w,mod6,'w'),grid,title('c.a.p.1,2,3,4,5,6'),xlabel
('pulsatia'),ylabel('Ampl');pause
    [mod7,arg7]=bode(num7,den7,w);
    semilogx(w,mod7,'r'),grid,title('c.a.p.1,2,3,4,5,6,7'),xlabel
('pulsatia'),ylabel('Ampl');pause
    hold off
  end     %ifk100=1

  if k100==2
    [mod1,arg1]=bode(num1,den1,w);
    semilogx(w,mod1,'r'),grid,title('c.a.p.1'),              xlabel
('pulsatia'), ylabel ('Ampl');pause
  end     %ifk100=2

  if k100==3
    [mod2,arg2]=bode(num2,den2,w);
    semilogx(w,mod2,'b'),grid,title('c.a.p.2'),xlabel
('pulsatia'),ylabel('Ampl');pause
  end     %ifk100=3

  if k100==4
    [mod3,arg3]=bode(num3,den3,w);
    semilogx(w,mod3,'y'),grid,title('c.a.p.3'),xlabel
('pulsatia'),ylabel('Ampl');pause
  end      %ifk100=4

  if k100==5
    [mod4,arg4]=bode(num4,den4,w);
    semilogx(w,mod4,'m'),grid,title('c.a.p.4'),xlabel
('pulsatia'),ylabel('Ampl');pause
  end      %ifk100=5

  if k100==6
    [mod5,arg5]=bode(num5,den5,w);
    semilogx(w,mod5,'c'),grid,title('c.a.p.5'),xlabel
('pulsatia'),ylabel('Ampl');pause
  end      %ifk100=6
  if k100==7
    [mod6,arg6]=bode(num6,den6,w);
    semilogx(w,mod6,'w'),grid,title('c.a.p.6'),xlabel
('pulsatia'),ylabel('Ampl');pause
  end      %ifk100=7

  if k100==8
    [mod7,arg7]=bode(num7,den7,w);
    semilogx(w,mod7,'r'),grid,title('c.a.p.7'),xlabel
('pulsatia'),ylabel('Ampl');pause
  end      %ifk100=8

  if k100==9
    hold off
    [mod1,arg1]=bode(num1,den1,w);
    semilogx(w,mod1,'r'),grid,title('c.a.p.1'),xlabel
('pulsatia'),ylabel('Ampl');
    hold on
    [mod2,arg2]=bode(num2,den2,w);
    semilogx(w,mod2,'b'),grid,title('c.a.p.1,2'),xlabel
('pulsatia'),ylabel('Ampl');
    [mod3,arg3]=bode(num3,den3,w);
    semilogx(w,mod3,'y'),grid,title('c.a.p.1,2,3'),xlabel
('pulsatia'),ylabel('Ampl');
    [mod4,arg4]=bode(num4,den4,w);
    semilogx(w,mod4,'m'),grid,title('c.a.p.1,2,3,4'),xlabel
('pulsatia'),ylabel('Ampl');
    [mod5,arg5]=bode(num5,den5,w);
    semilogx(w,mod5,'c'),grid,title('c.a.p.1,2,3,4,5'),xlabel
('pulsatia'),ylabel('Ampl');
    [mod6,arg6]=bode(num6,den6,w);
    semilogx(w,mod6,'w'),grid,title('c.a.p.1,2,3,4,5,6'),
xlabel('pulsatia'),ylabel('Ampl');
    [mod7,arg7]=bode(num7,den7,w);
    semilogx(w,mod7,'r'),grid,title('c.a.p.1,2,3,4,5,6,7'),
xlabel('pulsatia'),ylabel('Ampl');pause
    hold off
  end     %ifk100=9

  if k100==10
     t100=1;
  end     %ifk100=10
 end     %while_t100
end                 %ifk11=8

if k11==9
     hold off
     w=logspace(1,2.5,100);
     [mod1,arg1]=bode(num1,den1,w);
     modd1=20*log10(mod1);
     semilogx(w,modd1,'w'),grid,title('c.a.p.1'),xlabel
('pulsatia'),ylabel('Ampl');
    hold on
    [mod2,arg2]=bode(num2,den2,w);
    modd2=20*log10(mod2);
    semilogx(w,modd2,'w'),grid,title('c.a.p.2'),xlabel
('pulsatia'),ylabel('Ampl');
    [mod3,arg3]=bode(num3,den3,w);
    modd3=20*log10(mod3);
    semilogx(w,modd3,'w'),grid,title('c.a.p.3'),xlabel
('pulsatia'),ylabel('Ampl');
    [mod4,arg4]=bode(num4,den4,w);
    modd4=20*log10(mod4);
    semilogx(w,modd4,'w'),grid,title('c.a.p.4'),xlabel
('pulsatia'),ylabel('Ampl');
    [mod5,arg5]=bode(num5,den5,w);
    modd5=20*log10(mod5);
    semilogx(w,modd5,'w'),grid,title('c.a.p.5'),xlabel
('pulsatia'),ylabel('Ampl');
    [mod6,arg6]=bode(num6,den6,w);
    modd6=20*log10(mod6);
    semilogx(w,modd6,'w'),grid,title('c.a.p.6'),xlabel
('pulsatia'),ylabel('Ampl');
    [mod7,arg7]=bode(num7,den7,w);
    modd7=20*log10(mod7);
    semilogx(w,modd7,'w'),grid,title('c.a.p.7'),xlabel
('pulsatia'),ylabel('Ampl');pause
    hold off
end                 %ifk11=9

if k11==10;
  t11=1;
 end;                %ifk11=10
end;                %while_t11

%Generare semnal de intrare: semnal sinusoidal
%n=input('Dati lungimea secventei  n=');
%F0=input('Dati amplitudinea fortei F0=');
tfin=0;
while tfin==0
 t2=0;
 while t2==0
    k2=menu('SEMNAL   INTRARE','SINUSOIDAL','IMPULS',   'TREAPTA',
'CONTINUE');
    if k2==1
     k6=menu('init','de la tastatura','implicite');
     if k6==1
         n=input('Dati lungimea secventei  n=');
         x0=input('Dati amplitudinea miscarii x0=');
         F0=k8*x0;
         k7=menu('Dati','pulsatia','frecventa');
     if k7==1
       omega=input('Dati pulsatia:');
       niu=omega/(2*pi);        %frecventa
     end;                                 %ifk7=1
     if k7==2
       niu=input('Dati frecventa, in Hz:  ');
     end;                                 %ifk7=2
  end;                  %ifk6=1
  if k6==2
      n=3;
      x0=0.01;
      F0=k8*0.01;
      niu=10;
  end;                  %ifk6=2
  T=1/niu;              %calculul perioadei oscilatiilor
  i=0:0.01:n;
  u=x0*sin(2*pi*i./T);
  p=u';
  plot(0:0.01:n,u),grid,title('Semnal intrare  xf - sinus');
  xlabel('t'),ylabel('x0');
  pause;
  [y,t]=lsim(Am,Bm,Cm,Dm,p,0:0.01:n);
  [y1,t]=lsim(Am1,Bm1,Cm1,Dm1,p,0:0.01:n);
  [y2,t]=lsim(Am2,Bm2,Cm2,Dm2,p,0:0.01:n);
  [y3,t]=lsim(Am3,Bm3,Cm3,Dm3,p,0:0.01:n);
  [y4,t]=lsim(Am4,Bm4,Cm4,Dm4,p,0:0.01:n);
  [y5,t]=lsim(Am5,Bm5,Cm5,Dm5,p,0:0.01:n);
  [y6,t]=lsim(Am6,Bm6,Cm6,Dm6,p,0:0.01:n);
  [y7,t]=lsim(Am7,Bm7,Cm7,Dm7,p,0:0.01:n);
end;                    %ifk2=1

 if k2==2
  k6=menu('VALORI INPUT','de la tastatura','implicite');
  if k6==1
      n=input('Dati lungimea secventei  n=');
      F0=input('Dati amplitudinea fortei F0=');
      b=input('Dati lungimea impulsului b<n,   b=');
  end;                  %ifk6=1
  if k6==2
      n=10;
      F0=200;
      b=1;
  end;                  %ifk6=2
  u=zeros(1,n);
     for i=1:0.01:b
       u(i)=F0;
       p=u';
       end;                   %for
  plot(1:n,u),grid,title('FORTA PERTURBATOARE  - impuls');
  xlabel('t'),ylabel('F');
  pause;
```

```
[y,t]=lsim(Am,Bm,Cm,Dm,p,1:n);
[y1,t]=lsim(Am1,Bm1,Cm1,Dm1,p,1:n);
[y2,t]=lsim(Am2,Bm2,Cm2,Dm2,p,1:n);
[y3,t]=lsim(Am3,Bm3,Cm3,Dm3,p,1:n);
[y4,t]=lsim(Am4,Bm4,Cm4,Dm4,p,1:n);
[y5,t]=lsim(Am5,Bm5,Cm5,Dm5,p,1:n);
[y6,t]=lsim(Am6,Bm6,Cm6,Dm6,p,1:n);
[y7,t]=lsim(Am7,Bm7,Cm7,Dm7,p,1:n);
    end;                    %ifk2=2

    if k2==3
     k6=menu('init','de la tastatura','implicite');
     if k6==1
         n=input('Dati lungimea secventei  n=');
         F0=input('Dati amplitudinea fortei F0=');
     end;                %ifk6=1
     if k6==2
         n=10;
         F0=1;
     end;                %ifk6=2
     for i=0:n
      u(i)=F0;
      p=u';
     end;                %for
     plot(0:n,u),grid,title('FORTA PERTURBATOARE - treapta');
     xlabel('t'),ylabel('F');
     pause;
     [y,t]=lsim(Am,Bm,Cm,Dm,p,0:n);
     [y1,t]=lsim(Am1,Bm1,Cm1,Dm1,p,0:n);
     [y2,t]=lsim(Am2,Bm2,Cm2,Dm2,p,0:n);
     [y3,t]=lsim(Am3,Bm3,Cm3,Dm3,p,0:n);
     [y4,t]=lsim(Am4,Bm4,Cm4,Dm4,p,0:n);
     [y5,t]=lsim(Am5,Bm5,Cm5,Dm5,p,0:n);
     [y6,t]=lsim(Am6,Bm6,Cm6,Dm6,p,0:n);
     [y7,t]=lsim(Am7,Bm7,Cm7,Dm7,p,0:n);
    end;                    %ifk2=3

    if k2==4
     t2=1;
    end;                    %ifk2=4
end;                        %while_t2

%obtinerea raspunsului; simulare
%afisarea grafica a raspunsului
t3=0;
while t3==0
    k3=menu('AFISARE','un
parametru','all','all1','all2','all3','all4','REVENIRE','EXIT');
    if k3==1
     t4=0;
     while t4==0
       k4=menu('PARAMETRI','z1','z2','z3','z4','z5','z6',     'z7',
'Quit');
         if k4==1
          plot(y1);grid;xlabel('t');ylabel('z1');
          pause;
         end;            %ifk4=1
         if k4==2
          plot(y2),grid,xlabel('t'),ylabel('z2');
          pause;
         end;            %ifk4=2
         if k4==3
          plot(y3),grid,xlabel('t'),ylabel('z3');
          pause;
         end;            %ifk4=3
         if k4==4
          plot(y4);grid;xlabel('t');ylabel('z4');
          pause;
         end;            %ifk4=4
         if k4==5
          plot(y5);grid;xlabel('t');ylabel('z5');
          pause;
         end;            %ifk4=5
         if k4==6
          plot(y6);grid;xlabel('t');ylabel('z6');
          pause;
         end;            %ifk4=6
         if k4==7
          plot(y7);grid;xlabel('t');ylabel('z7');
          pause;
         end;            %ifk4=7
         if k4==8
          t4=1;
         end;            %ifk4=8
     end;                %while_t4
    end;                    %ifk3=1
    if k3==2
     plot(y);grid;xlabel('t');ylabel('y');
     pause;
    end;                    %ifk3=2

    if k3==3
    subplot(111),plot(y1,'w'),grid,xlabel('t'),ylabel('z1'),pause;
     plot(y2,'w'),grid,xlabel('t'),ylabel('z2');pause;
     plot(y3,'w'),grid,xlabel('t'),ylabel('z3');pause;
     plot(y4,'w'),grid,xlabel('t'),ylabel('z4');pause;
     plot(y5,'w'),grid,xlabel('t'),ylabel('z5');pause;
     plot(y6,'w'),grid,xlabel('t'),ylabel('z6');pause;
     plot(y7,'w'),grid,xlabel('t'),ylabel('z7');pause;

    end;                    %ifk3=3
    if k3==4
     subplot(221),plot(y1,'w'),grid;xlabel('t');ylabel('z1');
pause
     subplot(222),plot(y2,'w'),grid;xlabel('t');ylabel('z2');
pause
     subplot(223),plot(y3,'w'),grid;xlabel('t');ylabel('z3');
pause
     subplot(224),plot(y4,'w'),grid;xlabel('t');ylabel('z4');
pause
     clg;
     subplot(221),plot(y5,'w'),grid;xlabel('t');ylabel('z5');
pause
     subplot(222),plot(y6,'w'),grid;xlabel('t');ylabel('z6');
pause
     subplot(223),plot(y7,'w'),grid;xlabel('t');ylabel('z7');
     subplot(111);
     clg
end;                    %ifk3=4

    if k3==5
     subplot(1,1,1);
     hold off;
     plot(u,'w');
     hold on
     plot(y);grid;xlabel('t');ylabel('y');
     pause;
     hold off;
    end;                    %ifk3=5

    if k3==6
     hold off
     hold on;
     subplot(111)
     plot(u,'w');
     pause
     hold on
     plot(y1,'m'),grid,xlabel('t'),ylabel('..+z1'),pause;
     pause
     plot(y2,'r'),grid,xlabel('t'),ylabel('..+z2');pause;
     plot(y3,'r'),grid,xlabel('t'),ylabel('..+z3');pause;
     plot(y4,'c'),grid,xlabel('t'),ylabel('..+z4');pause;
     plot(y5,'y'),grid,xlabel('t'),ylabel('..+z5');pause;
     plot(y6,'b'),grid,xlabel('t'),ylabel('..+z6');pause;
     plot(y7,'w'),grid,xlabel('t'),ylabel('..+z7');pause;
     hold off
    end;                    %ifk3=6

    if k3==7
     t3=1;
     end   %ifk3=7

    if k3==8
     t3=1;
     tfin=1;
    end;                    %ifk3=8
 end;                        %while_t3
end;   %while_tfin

cla;
disp(''),disp(''),disp(''),disp(''),disp(''),disp(''),disp(''),dis
p('');
disp(''),disp('');
disp('              VA MULTUMIM CA ATI FOLOSIT ACEST PROGRAM');
disp(''),disp(''),disp(''),disp(''),disp(''),disp('');
disp('                                                 prof.coord.:
doctorand:');
disp('                    prof.univ.dr.ing.Liviu          BRINDEU
ing.Cornel POPA');
disp(''),disp(''),disp(''),disp(''),disp(''),disp('');
end                            %program
```

ANEXA 7.2

În continuare este redat codul sursă al programului *elasez.pas*. Pentru scrierea acestui program a fost folosit limbajul de programare Pascal.

```
program estimare_parametri_model;
uses crt,printer;
var men1,men2:integer;
    m,s1,m1,m2,m3,m4,m5,m6,m7,r:real;
    pr1,pr2,pr3,pr4,pr5,pr6,pr7,pr:real;
    fn1,fn2,fn3,fn4,fn5,fn6,fn7,a,b,c:real;
    omega1,omega2,omega3,omega4,omega5,omega6,omega7:real;
    k1,k2,k3,k4,k5,k6,k7,k8:real;
    ch,vm,vp,vf,vimp,cp,st,im:char;

procedure tasta1;
     begin
          ch:=readkey;
     end;

procedure antet;
     begin
          textbackground(3);
          clrscr;
          textbackground(1);
          textcolor(3);
          gotoxy(20,10);
          writeln('   M  O  D  E  L     M  E  C  A  N  I  C');
          gotoxy(10,12);
          writeln('al corpului omenesc stand asezat platform†
vibranta');
          writeln;
          delay(2050);
          clrscr;
          textbackground(3);
          textcolor(1);
          gotoxy(12,3);
          writeln('  D E T E R M I N A R E    P A R A M E T R I');
          writeln;
          writeln;
          tasta1;
     end;

procedure tasta;
     begin
          gotoxy(23,20);
          writeln('>> APASATI ORICE TASTA PENTRU A COMTINUA! <<');
          ch:=readkey;
     end;

procedure meniu1;
     begin
          writeln('       D A T E   I N I T I A L E');
          writeln('       ~~~~~~~~~~~~~~~~~~~~~~~~~');
          writeln('  Tasteaz†:');
          writeln;
          writeln('     1 ..... pentru folosirea datelor
implicite');
          writeln;
          writeln('     2 ..... pentru a introduce datele de la
tastatur†');
          writeln;
          readln(men1);
          writeln;
          writeln;
     end;

procedure meniu2;
     begin
          writeln('    Tasteaz†:');
          writeln;
          writeln('      1 ..... Dati masa total† a corpului');
          writeln;
          writeln('      2 ..... Dati masele celor 7 elemente
componente');
          writeln;
          readln(men2);
          writeln;
     end;

procedure masa1;
     begin
          clrscr;
          writeln('Dati masa total† a corpului, exprimat† Œn kg
!');
          write('        m=');
          readln(m);
          writeln;
          m1:=6.82*m/100;
          m2:=8.52*m/100;
          m3:=40.91*m/100;
          m4:=1.7*m/100;
          m5:=0.57*m/100;
          m6:=7.38*m/100;
          m7:=34.09*m/100;
          write('Doresti s† vezi valorile maselor componente?
(D/N)');
          readln(vm);
          writeln;
```

```
          if (vm='d') or (vm='D') then
             begin
                  writeln;
                  writeln('Masele elementelor modelului sunt:');
                  writeln;
                  writeln('     m1=',m1:9:3,' kg');
                  writeln('     m2=',m2:9:3,' kg');
                  writeln('     m3=',m3:9:3,' kg');
                  writeln('     m4=',m4:9:3,' kg');
                  writeln('     m5=',m5:9:3,' kg');
                  writeln('     m6=',m6:9:3,' kg');
                  writeln('     m7=',m7:9:3,' kg');
                  tasta;
             end;
     end;

procedure masa2;
     begin
          clrscr;
          writeln('Dati masele elementelor componente ale
modelului, Œn kg !');
          writeln;
          write('      m1=');readln(m1);
          write('      m2=');readln(m2);
          write('      m3=');readln(m3);
          write('      m4=');readln(m4);
          write('      m5=');readln(m5);
          write('      m6=');readln(m6);
          write('      m7=');readln(m7);
          writeln;
          m:=m1+m2+m3+m4+m5+m6+m7;
          delay(850);
          writeln('Masa total† a corpului omenesc este:
m=',m:9:3,' kg');
          writeln;
          tasta1;
     end;

procedure frecventa;
     begin
          writeln;
          writeln('Doresti s† dai valorile frecventelor de
rezonant† ale');
          write('    maselor componente ale modelului mecanic ?
(D/N)  ');
          readln(vf);
          writeln;
          if (vf='d') or (vf='D') then
             begin
                 writeln('Dati frecventele de rezonant†, Œn Hz!');
                 writeln;
                 write('             fn1:=');readln(fn1);
                 write('             fn2:=');readln(fn2);
                 write('             fn3:=');readln(fn3);
                 write('             fn4:=');readln(fn4);
                 write('             fn5:=');readln(fn5);
                 write('             fn6:=');readln(fn6);
                 write('             fn7:=');readln(fn7);
                 writeln;
                 write('Doresti s† vezi valorile pulsatiilor?
(D/N)  ');
                 readln(vp);
                 writeln;
                 if (vp='d') or (vp='D') then
                    begin
                         writeln;
                         writeln('Pulsatiile aferente elementelor
modelului sunt:');
                         writeln;
                         omega1:=2*pi*fn1;
                         writeln('     omega1=',omega1:7:2,'
rad/s');
                         omega2:=2*pi*fn2;
                         writeln('     omega2=',omega2:7:2,'
rad/s');
                         omega3:=2*pi*fn3;
                         writeln('     omega3=',omega3:7:2,'
rad/s');
                         omega4:=2*pi*fn4;
                         writeln('     omega4=',omega4:7:2,'
rad/s');
                         omega5:=2*pi*fn5;
                         writeln('     omega5=',omega5:7:2,'
rad/s');
                         omega6:=2*pi*fn6;
                         writeln('     omega6=',omega6:7:2,'
rad/s');
                         omega7:=2*pi*fn7;
                         writeln('     omega7=',omega7:7:2,'
rad/s');
                         readln;
                    end;
             end;
          if (vf<>'d') or (vf<>'D') then
             begin

          fn1:=15;  fn2:=21;  fn3:=4.5;  fn4:=5;
          fn5:=10;  fn6:=3;  fn7:=7.5;

          end;
     end;
```

```
procedure date_init;
     begin
          m:=80;
          m1:=6.82*m/100;
          m2:=8.52*m/100;
          m3:=40.91*m/100;
          m4:=1.7*m/100;
          m5:=0.57*m/100;
          m6:=7.38*m/100;
          m7:=34.09*m/100;

          fn1:=15;  fn2:=21;  fn3:=4.5;  fn4:=5;
          fn5:=10;  fn6:=3;  fn7:=7.5;

     end;

procedure calcul_afisare;
     begin
          clrscr;
          textbackground(1);
          textcolor(yellow);
          gotoxy(10,4);
          writeln('P A R A M E T R I I   M  O D E L U L U I  M E C
A N I C:');
          gotoxy(10,5);

writeln('ßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßß'
);
          writeln;
          k1:=m1*sqr(2*pi*fn1);
          k4:=m4*sqr(2*pi*fn4)/2;
          k3:=m3*sqr(2*pi*fn3)-k4;
          k6:=m6*sqr(2*pi*fn6)/2;
          fn5:=1/(2*pi)*sqrt((k4+k6)/m5);
          k2:=m2*sqr(2*pi*fn2)-k1-k3;
          k8:=m7*sqr(2*pi*fn7)-k6-k2;
          k5:=k4;
          k7:=k6;

          writeln('  Valorile parametrilor elementelor elastice
sunt:');
          writeln('     k1=',k1:9:2,' N/m');
          writeln('     k2=',k2:9:2,' N/m');
          writeln('     k3=',k3:9:2,' N/m');
          writeln('     k4=',k4:9:2,' N/m');
          writeln('     k5=',k5:9:2,' N/m');
          writeln('     k6=',k6:9:2,' N/m');
          writeln('     k7=',k7:9:2,' N/m');
          writeln('     k8=',k8:9:2,' N/m');
          writeln; writeln;
          m:=m1+m2+m3+m4+m5+m6+m7;
          pr1:=m1*100/round(m);
          pr2:=m2*100/round(m);
          pr3:=m3*100/round(m);
          pr4:=m4*100/round(m);
          pr5:=m5*100/round(m);
          pr6:=m6*100/round(m);
          pr7:=m7*100/round(m);
          pr:=pr1+pr2+pr3+pr4+pr5+pr6+pr7;
          writeln('  Masele elementelor componente ale corpului
sunt:');
          writeln('      m1=',m1:9:2,' kg
',pr1:5:1,'%');
          writeln('      m2=',m2:9:2,' kg
',pr2:5:1,'%');
          writeln('      m3=',m3:9:2,' kg
',pr3:5:1,'%');
          writeln('      m4=',m4:9:2,' kg
',pr4:5:1,'%');
          writeln('      m5=',m5:9:2,' kg
',pr5:5:1,'%');
          writeln('      m6=',m6:9:2,' kg
',pr6:5:1,'%');
          writeln('      m7=',m7:9:2,' kg
',pr7:5:1,'%');
          writeln;
          m:=m1+m2+m3+m4+m5+m6+m7;
          writeln('  Masa total† a corpului omenesc este:
m=',round(m),' kg');
          tasta1;
end;

procedure tiparire_imprimanta;
     begin
          clrscr;
          writeln('    Vrei s† tip†resti rezultatele la imprimant†
?  ');
          writeln;
          writeln('Tasteaz†:');
          writeln;
          writeln('      1 ..... pentru tip†rire la imprimant†');
          writeln;
          writeln('      2 ..... pentru NU');
          writeln;
          readln(vimp);
          writeln;
          if (vimp='1') then
             begin
                clrscr;
                textbackground(7);
                textcolor(5);
                gotoxy(10,10);
                writeln('Esti sigur ct vrei st tiptresti la
imprimantt valorile');
                write('              parametrilor ? (D/N)  ');
                readln(cp);
                writeln;
                if (cp='d') or (cp='D') then
                  begin
                     textbackground(5);
                     clrscr;
                     textbackground(1);
                     textcolor(135);
                     gotoxy(30,10);
                     writeln('T I P A R I R E');
                     gotoxy(28,12);
                     writeln('*******************');
                     gotoxy(28,8);
                     writeln('*******************');
                     writeln(lst,'  P A R A M E T R I I   M  O D E
L U L U I  M E C A N I C:');
                     writeln(lst,'
ßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßß');
                     writeln(lst,'');
                     writeln(lst,'     k1=',k1:9:2,' N/m');
                     writeln(lst,'     k2=',k2:9:2,' N/m');
                     writeln(lst,'     k3=',k3:9:2,' N/m');
                     writeln(lst,'     k4=',k4:9:2,' N/m');
                     writeln(lst,'     k5=',k5:9:2,' N/m');
                     writeln(lst,'     k6=',k6:9:2,' N/m');
                     writeln(lst,'     k7=',k7:9:2,' N/m');
                     writeln(lst,'     k8=',k8:9:2,' N/m');

                     writeln(lst,''); writeln(lst,'');
                     writeln(lst,'  Masele elementelor componente
ale corpului sunt:');
                     writeln(lst,'');
                     writeln(lst,'      m1=',m1:9:2,' kg
',m1*100/round(m):5:1,' %');
                     writeln(lst,'      m2=',m2:9:2,' kg
',m2*100/round(m):5:1,' %');
                     writeln(lst,'      m3=',m3:9:2,' kg
',m3*100/round(m):5:1,' %');
                     writeln(lst,'      m4=',m4:9:2,' kg
',m4*100/round(m):5:1,' %');
                     writeln(lst,'      m5=',m5:9:2,' kg
',m5*100/round(m):5:1,' %');
                     writeln(lst,'      m6=',m6:9:2,' kg
',m6*100/round(m):5:1,' %');
                     writeln(lst,'      m7=',m7:9:2,' kg
',m7*100/round(m):5:1,' %');
                     writeln(lst,'');
                     m:=m1+m2+m3+m4+m5+m6+m7;
                     writeln(lst,'  Masa total† a corpului omenesc
este: m=',round(m),' kg');
                     readln;
                     textcolor(7);
                  end;
             end;
     end;

procedure calcul;
    begin

          m1:=6.82*m/100;
          m2:=8.52*m/100;
          m3:=40.91*m/100;
          m4:=1.7*m/100;
          m5:=0.57*m/100;
          m6:=7.38*m/100;
          m7:=34.09*m/100;

          fn1:=15;  fn2:=21;  fn3:=4.5;  fn4:=5;
          fn5:=10;  fn6:=3;  fn7:=7.5;

          k1:=m1*sqr(2*pi*fn1);
          k4:=m4*sqr(2*pi*fn4)/2;
          k3:=m3*sqr(2*pi*fn3)-k4;
          k6:=m6*sqr(2*pi*fn6)/2;
          fn5:=1/(2*pi)*sqrt((k4+k6)/m5);
          k2:=m2*sqr(2*pi*fn2)-k1-k3;
          k8:=m7*sqr(2*pi*fn7)-k6-k2;
          k5:=k4;
          k7:=k6;

end;

procedure param;
    begin
         clrscr;
         writeln; writeln; writeln;
         writeln('                Variatia parametrilor ki functie
de masa totala');
         writeln('                --------------------------------
----------------');
         writeln;
         writeln('
=================================================================
=========');
         writeln('   m[kg]  k1[N/m]   k2[N/m]   k3[N/m]   k4[N/m]
k5[N/m]   k6[N/m]   k7[N/m]   k8[N/m]');
         writeln('
=================================================================
=========');
```

```
         m:=50;
         while m<=100 do
               begin
                  calcul;
                  writeln(' ',m:7:2,' ',k1:9:2,' ',k2:9:2,'
',k3:9:2,' ',k4:9:2,' ',k5:9:2,' ',k6:9:2,' ',k7:9:2,' ',k8:9:2);
                  m:=m+6;
               end;
         writeln('
==================================================================
=========');
         readln;
    end;

procedure param_im;
    begin
         writeln(lst);writeln(lst);
         writeln(lst,'            Variatia parametrilor ki functie
de masa totala');
         writeln(lst,'            --------------------------------
----------------');
         writeln(lst);
         writeln(lst,'
==================================================================
=========');
         writeln(lst,'   m[kg]  k1[N/m]   k2[N/m]   k3[N/m]
k4[N/m]   k5[N/m]   k6[N/m]   k7[N/m]');
         writeln(lst,'
==================================================================
=========');
         m:=50;
         while m<=100 do
               begin
                  calcul;
                  writeln(lst,' ',m:7:2,' ',k1:9:2,' ',k2:9:2,'
',k3:9:2,' ',k4:9:2,' ',k5:9:2,' ',k6:9:2,
                  ' ',k7:9:2,' ',k8:9:2);
                  m:=m+6;
               end;
    end;


procedure param1;
    begin
         clrscr;
         writeln; writeln; writeln;
         writeln('            Variatia maselor partilor corpului
functie de masa totala');
         writeln('            -----------------------------------
----------------------');
         writeln;
         writeln('
==================================================================
=========');
         writeln('   m[kg]    m1[kg]    m2[kg]    m3[kg]    m4[kg]
m5[kg]    m6[kg]    m7[kg]');
         writeln('
==================================================================
=========');
         m:=50;
         while m<=100 do
               begin
                  calcul;
                  writeln(' ',m:7:2,' ',m1:9:2,' ',m2:9:2,'
',m3:9:2,' ',m4:9:2,' ',m5:9:2,' ',m6:9:2,' ',m7:9:2);
                  m:=m+6;
               end;
         writeln('
==================================================================
=========');
         readln;
    end;

procedure param_im1;
    begin
         writeln(lst);writeln(lst);
         writeln(lst,'        Variatia maselor partilor corpului
functie de masa totala');
         writeln(lst,'        -----------------------------------
----------------------');
         writeln(lst);
         writeln(lst,'   m[kg]    m1[kg]    m2[kg]    m3[kg]
m4[kg]    m5[kg]    m6[kg]    m7[kg]');
         writeln(lst,'
==================================================================
=========');
         m:=50;
         while m<=100 do
               begin
                  calcul;
                  writeln(lst,' ',m:7:2,' ',m1:9:2,' ',m2:9:2,'
',m3:9:2,' ',m4:9:2,' ',m5:9:2,' ',m6:9:2,' ',m7:9:2);
                  m:=m+6;
               end;
    end;


begin
     antet;
     meniu1;
     if (men1=1) then
          date_init;
     if (men1=2) then
          begin
               meniu2;
               if (men2=1) then
                    masa1;
               if (men2=2) then
                    masa2;
          end;
     frecventa;
     calcul_afisare;
     tiparire_imprimanta;
     clrscr;
     writeln;writeln;writeln;writeln;writeln;writeln;
     writeln(' Doriti sa consultati niste date statistice?'  );
     writeln;
     write('        [d/n]');
     readln(st);
     if (st='d') or (st='D') then
        begin
             param;
             writeln;writeln;
             writeln('fn5=',fn5:7:2);
             writeln(' Doriti sa le tipariti la imprimanta?  ');
             write('        [d/n]');
             readln(im);
             if (im='d') or (im='D') then
                param_im;
             param1;
             writeln;writeln;
             writeln(' Doriti sa le tipariti la imprimanta?  ');
             write('        [d/n]');
             readln(im);
             if (im='d') or (im='D') then
                param_im1;
        end;
end.
```

ANEXA 7.3

Listingul obținut prin rularea programului *elasasez.exe*:

```
P A R A M E T R I I   M O D E L U L U I   M E C A N I C:


    k1= 48463.71 N/m
    k2= 44709.86 N/m
    k3= 25492.87 N/m
    k4=   671.13 N/m
    k5=   671.13 N/m
    k6=  1048.86 N/m
    k7=  1048.86 N/m
    k8= 14803.14 N/m


Masele elementelor componente ale corpului sunt:

     m1=      5.46 kg                         6.8 %
     m2=      6.82 kg                         8.5 %
     m3=     32.73 kg                        40.9 %
     m4=      1.36 kg                         1.7 %
     m5=      0.46 kg                         0.6 %
     m6=      5.90 kg                         7.4 %
     m7=     27.27 kg                        34.1 %

Masa totală a corpului omenesc este: m=80 kg


          Variatia parametrilor ki functie de masa totala
          -----------------------------------------------

=========================================================================
m[kg]  k1[N/m]   k2[N/m]   k3[N/m]   k4[N/m]   k5[N/m]   k6[N/m]   k7[N/m]
   k8[N/m]
=========================================================================
50.00  30289.82  27943.66  15933.05   419.46    419.46    655.54    655.5
    9251.96
56.00  33924.59  31296.90  17845.01   469.79    469.79    734.20    734.2
   10362.20
62.00  37559.37  34650.14  19756.98   520.13    520.13    812.87    812.8
   11472.44
68.00  41194.15  38003.38  21668.94   570.46    570.46    891.53    891.5
   12582.67
74.00  44828.93  41356.62  23580.91   620.80    620.80    970.20    970.2
   13692.91
80.00  48463.71  44709.86  25492.87   671.13    671.13   1048.86   1048.8
   14803.14
86.00  52098.48  48063.10  27404.84   721.47    721.47   1127.53   1127.5
   15913.38
92.00  55733.26  51416.34  29316.80   771.80    771.80   1206.19   1206.1
   17023.61
98.00  59368.04  54769.58  31228.77   822.14    822.14   1284.86   1284.8
   18133.85
```

Variatia maselor partilor corpului functie de masa totala

m[kg]	m1[kg]	m2[kg]	m3[kg]	m4[kg]	m5[kg]	m6[kg]	m7[kg
50.00	3.41	4.26	20.46	0.85	0.28	3.69	17.05
56.00	3.82	4.77	22.91	0.95	0.32	4.13	19.09
62.00	4.23	5.28	25.36	1.05	0.35	4.58	21.14
68.00	4.64	5.79	27.82	1.16	0.39	5.02	23.18
74.00	5.05	6.30	30.27	1.26	0.42	5.46	25.23
80.00	5.46	6.82	32.73	1.36	0.46	5.90	27.27
86.00	5.87	7.33	35.18	1.46	0.49	6.35	29.32
92.00	6.27	7.84	37.64	1.56	0.52	6.79	31.36
98.00	6.68	8.35	40.09	1.67	0.56	7.23	33.41

ANEXA 7.4

În continuare este redat codul sursă al programului *amortiz.pas*. Pentru scrierea acestui program a fost folosit limbajul de programare Pascal.

```
program estimare_parametri_model;
uses crt,printer;
var men1,men2:integer;
    m,s1,m1,m2,m3,m4,m5,m6,m7,r:real;
    pr1,pr2,pr3,pr4,pr5,pr6,pr7,pr:real;
    fn1,fn2,fn3,fn4,fn5,fn6,fn7,a,b,c:real;
    n1,n2,n3,n4,n5,n6,n7,n8:real;
    omega1,omega2,omega3,omega4,omega5,omega6,omega7:real;
    k1,k2,k3,k4,k5,k6,k7,k8:real;
        k1p,k2p,k3p,k4p,k5p,k6p,k7p,k8p:real;
    c1,c2,c3,c4,c5,c6,c7,c8:real;
    c1cr,c2cr,c3cr,c4cr,c5cr,c6cr,c7cr,c8cr:real;
    alfa2,beta2,gamma2,alfa3,beta3,gamma3,alfa7,beta7,gamma7:real;
    niu2,niu3,niu7:real;
    p5,ppr,pse,c3cr1:real;
    ch,vm,vp,vf,vimp,cp,st,im:char;

procedure calcul;
begin
          clrscr;
          m:=80;
          m1:=6.82*m/100;
          m2:=8.52*m/100;
          m3:=40.91*m/100;
          m4:=1.7*m/100;
          m5:=0.57*m/100;
          m6:=7.38*m/100;
          m7:=34.09*m/100;
          fn1:=15;  fn2:=21;  fn3:=4.5;  fn4:=5;
          fn5:=10;  fn6:=3;  fn7:=7.5;
          n1:=10; n2:=10;  n3:=5;  n4:=25;
          n5:=25;  n6:=25;  n7:=25;
          omega1:=2*pi*fn1;
          omega2:=2*pi*fn2;
          omega3:=2*pi*fn3;
          omega4:=2*pi*fn4;
          omega5:=2*pi*fn5;
          omega6:=2*pi*fn6;
          omega7:=2*pi*fn7;
          k1:=m1*omega1*omega1/(1-n1*n1/10000);
          c1cr:=2*sqrt(k1*m1);
          c1:=n1*c1cr/100;
          k4:=m4*sqr(omega4)/(2*(1-sqr(n4)/10000));
          c4cr:=sqrt(2*k4*m4);
          c4:=n4*c4cr/100;
        k6:=m6*sqr(omega6)/(2*(1-sqr(n6)/10000));
        c6cr:=sqrt(2*k6*m6);
        c6:=n6*c6cr/100;
        alfa3:=1-sqr(n3)/10000;
        beta3:=n3*(n3-n4)*c4cr/10000;
        gamma3:=-sqr(m3*omega3)-sqr(n3-n4)*c4cr*c4cr/4/10000;
        niu3:=(-beta3+sqrt(beta3*beta34*alfa3*gamma3))/(2
*alfa3);
        k3:=niu3*niu3/m3-k4;
        c3cr:=2*niu3-c4cr;
        c3:=n3*c3cr/100;
        alfa2:=1-sqr(n2)/10000;
        beta2:=-n2/10000*((n1-n2)*c1cr+(n3-n4)*c3cr);
        gamma2:=-sqr(m2*omega2)-((n1-n2)*c1cr+(n3-
        n2)*c3cr)/4/10000;
        niu2:=(-beta2+sqrt(beta2*beta2-4*alfa2*gamma2))/2/alfa2;
        k2:=niu2*niu2/m2-k1-k3;
        c2cr:=abs(2*niu2-c1cr-c3cr);
        c2:=n2*c2cr/100;
        alfa7:=1-sqr(n7)/10000;
        beta7:=-n7/10000*((n7-n7)*c2cr+(n6-n7)*c6cr);
        gamma7:=-sqr(m7*omega7)-((n2-n7)*c2cr+(n6-n7)*c6cr)/4
/10000;
        niu7:=(-beta7+sqrt(beta7*beta7-4*alfa7*gamma7))/2/alfa7;
        k8:=niu7*niu7/m7-k2-k6;
        p5:=sqrt((k4+k6)/m5-sqr((c4+c6)/(2*m5)));
        fn5:=p5/2/pi;
        writeln('fn5=',fn5:5:2);
        k5:=k4;
        k7:=k6;
        c5cr:=c4cr;
        c7cr:=c6cr;
        c5:=n5*c5cr/100;
        c7:=n7*c7cr/100;
        writeln('Parametrii modelului mecanic');
        writeln('===========================');
        writeln;
        writeln('     k1=',k1:9:2,' N/m');
        writeln('     k2=',k2:9:2,' N/m');
        writeln('     k3=',k3:9:2,' N/m');
        writeln('     k4=',k4:9:2,' N/m');
        writeln('     k5=',k5:9:2,' N/m');
        writeln('     k6=',k6:9:2,' N/m');
        writeln('     k7=',k7:9:2,' N/m');
        writeln('     k8=',k8:9:2,' N/m');
        writeln; writeln;
        readln;
        writeln('     c1=',c1:9:2,' Ns/m');
        writeln('     c2=',c2:9:2,' Ns/m');
        writeln('     c3=',c3:9:2,' Ns/m');
        writeln('     c4=',c4:9:2,' Ns/m');
```

```
          writeln('     c5=',c5:9:2,' Ns/m');
          writeln('     c6=',c6:9:2,' Ns/m');
          writeln('     c7=',c7:9:2,' Ns/m');
          writeln; writeln;
          writeln('     Masele elementelor componente ale corpului
sunt:');
          writeln('      m1=',m1:9:2,' kg
',m1:5:1,'%');
          writeln('      m2=',m2:9:2,' kg
',m2:5:1,'%');
          writeln('      m3=',m3:9:2,' kg
',m3:5:1,'%');
          writeln('      m4=',m4:9:2,' kg
',m4:5:1,'%');
          writeln('      m5=',m5:9:2,' kg
',m5:5:1,'%');
          writeln('      m6=',m6:9:2,' kg
',m6:5:1,'%');
          writeln('      m7=',m7:9:2,' kg
',m7:5:1,'%');
          writeln;
          m:=m1+m2+m3+m4+m5+m6+m7;
          writeln('  Masa total† a corpului omenesc este:
m=',round(m),' kg');
          readln;
end;

procedure calculim;
    begin
          writeln(lst,'Parametrii modelului mecanic');
          writeln(lst,'============================');
          writeln(lst);
          writeln(lst,'     k1=',k1:9:2,' N/m');
          writeln(lst,'     k2=',k2:9:2,' N/m');
          writeln(lst,'     k3=',k3:9:2,' N/m');
          writeln(lst,'     k4=',k4:9:2,' N/m');
          writeln(lst,'     k5=',k5:9:2,' N/m');
          writeln(lst,'     k6=',k6:9:2,' N/m');
          writeln(lst,'     k7=',k7:9:2,' N/m');
          writeln(lst,'     k8=',k8:9:2,' N/m');
          writeln(lst); writeln(lst);
          readln;
          writeln(lst,'     c1=',c1:9:2,' Ns/m');
          writeln(lst,'     c2=',c2:9:2,' Ns/m');
          writeln(lst,'     c3=',c3:9:2,' Ns/m');
          writeln(lst,'     c4=',c4:9:2,' Ns/m');
          writeln(lst,'     c5=',c5:9:2,' Ns/m');
          writeln(lst,'     c6=',c6:9:2,' Ns/m');
          writeln(lst,'     c7=',c7:9:2,' Ns/m');
          writeln(lst); writeln(lst);
          writeln(lst,'  Masele elementelor componente ale
corpului sunt:');
          writeln(lst,'      m1=',m1:9:2,' kg
',m1:5:1,'%');
          writeln(lst,'      m2=',m2:9:2,' kg
',m2:5:1,'%');
          writeln(lst,'      m3=',m3:9:2,' kg
',m3:5:1,'%');
          writeln(lst,'      m4=',m4:9:2,' kg
',m4:5:1,'%');
          writeln(lst,'      m5=',m5:9:2,' kg
',m5:5:1,'%');
          writeln(lst,'      m6=',m6:9:2,' kg
',m6:5:1,'%');
          writeln(lst,'      m7=',m7:9:2,' kg
',m7:5:1,'%');
          writeln(lst);
          m:=m1+m2+m3+m4+m5+m6+m7;
          writeln(lst,'  Masa total† a corpului omenesc este:
m=',round(m),' kg');
          readln;
end;

procedure param;
    begin
         clrscr;
         writeln; writeln; writeln;
         writeln('                Variatia parametrilor ki functie
de masa totala');
         writeln('                --------------------------------
----------------');
         writeln;
         writeln('
==================================================================
=========');
         writeln('   m[kg]  k1[N/m]   k2[N/m]   k3[N/m]   k4[N/m]
k5[N/m]   k6[N/m]   k7[N/m]  k8[N/m]');
         writeln('
==================================================================
=========');
         m:=50;
         while m<=100 do
               begin
                  calcul;
                  writeln(' ',m:7:2,' ',k1:9:2,' ',k2:9:2,'
',k3:9:2,' ',k4:9:2,' ',k5:9:2,' ',k6:9:2,' ',k7:9:2,' ',k8:9:2);
                  m:=m+25;
               end;
         writeln('
==================================================================
=========');
```

```
         readln;
    end;

procedure param_im;
    begin
         writeln(lst);writeln(lst);
         writeln(lst,'            Variatia parametrilor ki functie
de masa totala');
         writeln(lst,'            --------------------------------
----------------');
         writeln(lst);
         writeln(lst,'
==================================================================
=========');
         writeln(lst,'   m[kg]  k1[N/m]   k2[N/m]   k3[N/m]
k4[N/m]   k5[N/m]   k6[N/m]   k7[N/m]');
         writeln(lst,'
==================================================================
=========');
         m:=50;
         while m<=100 do
               begin
                  calcul;
                  writeln(lst,' ',m:7:2,' ',k1:9:2,' ',k2:9:2,'
',k3:9:2,' ',k4:9:2,' ',k5:9:2,' ',k6:9:2,
                  ' ',k7:9:2,' ',k8:9:2);
                  m:=m+6;
               end;
    end;

procedure param1;
    begin
         clrscr;
         writeln; writeln; writeln;
         writeln('           Variatia maselor partilor corpului
functie de masa totala');
         writeln('           ------------------------------------
----------------------');
         writeln;
         writeln('
==================================================================
=========');
         writeln('   m[kg]    m1[kg]    m2[kg]    m3[kg]    m4[kg]
m5[kg]    m6[kg]    m7[kg]');
         writeln('
==================================================================
=========');
         m:=50;
         while m<=100 do
               begin
                  calcul;
                  writeln(' ',m:7:2,' ',m1:9:2,' ',m2:9:2,'
',m3:9:2,' ',m4:9:2,' ',m5:9:2,' ',m6:9:2,' ',m7:9:2);
                  m:=m+6;
               end;
         writeln('
==================================================================
=========');
         readln;
    end;

procedure param_im1;
    begin
         writeln(lst);writeln(lst);
         writeln(lst,'         Variatia maselor partilor corpului
functie de masa totala');
         writeln(lst,'         -----------------------------------
----------------------');
         writeln(lst);
         writeln(lst,'   m[kg]    m1[kg]    m2[kg]    m3[kg]
m4[kg]    m5[kg]    m6[kg]    m7[kg]');
         writeln(lst,'
==================================================================
=========');
         m:=50;
         while m<=100 do
               begin
                  calcul;
                  writeln(lst,' ',m:7:2,' ',m1:9:2,' ',m2:9:2,'
',m3:9:2,' ',m4:9:2,' ',m5:9:2,' ',m6:9:2,' ',m7:9:2);
                  m:=m+6;
               end;
    end;

begin
     calcul;
     readln;
             writeln(' Doriti sa le tipariti la imprimanta?  ');
             write('        [d/n]');
             readln(im);
             if (im='d') or (im='D') then
                calculim;
     readln;
end.
```

ANEXA 7.5

Listingul obținut prin rularea programului *amortiz.exe*:

```
Parametrii modelului mecanic
============================

     k1= 48953.24 N/m
     k2= 40634.65 N/m
     k3= 25526.83 N/m
     k4=   715.88 N/m
     k5=   715.88 N/m
     k6=  1118.79 N/m
     k7=  1118.79 N/m
     k8= 22845.87 N/m


     c1=   103.36 Ns/m
     c2=   107.14 Ns/m
     c3=    90.47 Ns/m
     c4=    11.03 Ns/m
     c5=    11.03 Ns/m
     c6=    28.73 Ns/m
     c7=    28.73 Ns/m


  Masele elementelor componente ale corpului sunt:
      m1=     5.46 kg                          5.5%
      m2=     6.82 kg                          6.8%
      m3=    32.73 kg                         32.7%
      m4=     1.36 kg                          1.4%
      m5=     0.46 kg                          0.5%
      m6=     5.90 kg                          5.9%
      m7=    27.27 kg                         27.3%

  Masa totalå a corpului omenesc este: m=80 kg
```

ANEXA 7.6

Conținutul fişierului de comenzi numit *amel.bat* este:

```
ECHO OFF

:RELUARE
CLS
ECHO
ECHO -------------------------------------------------------------
-----
ECHO !               K : fara amortizare
ECHO !               C : cu amortizare
ECHO !               X : EXIT
ECHO -------------------------------------------------------------
-----
ECHO
CHOICE /c:KCX Alegeti o comanda
IF ERRORLEVEL 3 GOTO FINAL
IF ERRORLEVEL 2 GOTO AM
IF ERRORLEVEL 1 GOTO ARC

:ARC
ECHO -------------------------------------------------------------
-----
ECHO !              A: programul e pe discheta
ECHO !              C: programul e pe harddisc
ECHO -------------------------------------------------------------
-----
ECHO
CHOICE /c:AC Optiunea:
IF ERRORLEVEL 2 GOTO ARC1
IF ERRORLEVEL 1 GOTO ARC2

:ARC2
CALL A:\ELASASEZ.EXE
PAUSE
GOTO RELUARE

:ARC1
CALL C:\LUCRU\ELASASEZ.EXE
PAUSE
GOTO RELUARE

:AM
ECHO -------------------------------------------------------------
-----
ECHO !              A: programul e pe discheta
ECHO !              C: programul e pe harddisk
ECHO -------------------------------------------------------------
-----
ECHO
CHOICE /c:AC Optiunea:
IF ERRORLEVEL 2 GOTO AM1
IF ERRORLEVEL 1 GOTO AM2

:AM2
CALL A:\AMORTIZ.EXE
GOTO RELUARE

:AM1
CALL C:\LUCRU\AMORTIZ.EXE
GOTO RELUARE

:FINAL
ECHO
******************************************************************
ECHO !
ECHO !     Daca nu a rulat programul e posibil ca sa nu fi fost
gasit
ECHO !
ECHO !     Deci:  - deschide fisierul paramet.bat pentru editare
ECHO !               (in NortonCommander cu <F4>), sau folosind
ECHO !               editorul Notepad (in Windows)
ECHO !
ECHO !            - modifica in mod corect calea la care se
gaseste
ECHO !               fisierul elastfin.exe si/sau fisierul
amortiz.exe
ECHO !
ECHO !            - salveaza modificarile aduse (in NC cu <F2>) si
ECHO !           inchide fisierul (in NC cu <F10>)
ECHO !
ECHO
******************************************************************

ECHO OFF
```

ANEXA 8.1

Codul sursă al programului de descriere a subsistemului “SOSEA”, *sos*1.*t*, realizat în mediul de programare Simnon, este prezentat în continuare:

```
     CONTINUOUS SYSTEM SUSPEN
     TIME t
     INPUT u
     OUTPUT h
     "nu are variabile de stare"
     w1=if t<1 then 0.01 else 0   "w1,w2 - variabile auxiliare
     w2=if t>2 and t<3 then 0.01 else 0
     h=u*(w1+w2)
     END
```

Codul sursă al programului de descriere a subsistemului “SUSPEN”, *sus*1.*t*, realizat în mediul de programare Simnon, este prezentat în continuare:

```
     CONTINUOUS SYSTEM SUSPEN        "Antet (fisier sus1.t)
     INPUT u                  "Declaratii  - intrari
     OUTPUT y1 y2                       - iesiri
     STATE x1 x2 x3 x4                              - variabile de
stare
     DERIVATE x1d x2d x3d x4d           - derivate
     x1d=x2                   "Ecuatii de simulare
     x2d=(c*x4+k1*x3-c*x2-k18x1)/m1
     x3d=x4
     x4d=(c*x2-(k1+k2)*x3-c*x4+k1*x1+k2*u)/m2
     y1=x1
     y2=x3
     m1:300                   "Parametri numerici
     m2:30
     c:200
     k1:14700
     k2:60300
     END
```

Codul sursă al programului de interconectare, *con*1.*t*, realizat în mediul de programare Simnon, este prezentat în continuare:

```
     CONECTING SYSTEM con1
     TIME t
     "Atribuire variabila de intrare pentru subsisteme
     u[SOSEA]=1
     u[SUSPEN]=h[SOSEA]
     END
```

Codul sursă al programului de simulare, *sim*1.*t*, realizat în mediul de programare Simnon, este prezentat în continuare:

```
MACRO sim1
SYST SOSEA SUSPEN CON1
"Activarea sistemelor care intervin
STORE d1[SUSPEN] d2[SUSPEN]
"Selectie variabile de memorat
SIMU 0 5                  "Simulare pe intervalul 0..4
"Se afiseaza rezultatele grafice in doua ferestre pe ecran,
"     una sub cealalta
SPLIT 2 1
ASHOW y1[SUSPEN]
ASHOW y2[SUSPEN]
END
```

ANEXA 8.2

Codul sursă al programului de descriere a sistemului “SUSPEN”, *sus*2.*t*, realizat în mediul de programare Simnon, este prezentat în continuare:

```
CONTINUOUS SYSTEM SUSPEN            "Antet (fisier sus1.t)
STATE x1 x2 x3 x4                   "variabile de stare
DER x1d x2d x3d x4d                 "derivate
x1d=x2                              "Ecuatii de simulare
x2d=(c*x4+k1*x3-c*x2-k1*x1)/m1
x3d=x4
x4d=(c*x2-(k1+k2)*x3-c*x4+k1*x1+k2*u)/m2
u=0.01*sin(2*3.14*f)
m1:300                              "Parametri numerici
m2:30
c:200
k1:14700
k2:60300
f:10
END
```

Codul sursă al programului de simulare, *sim*2.*t*, realizat în mediul de programare Simnon, este prezentat în continuare:

```
MACRO sim2
SYST SUSPEN
"Activarea sistemelor care intervin
STORE x1 x3
"Selectie variabile de memorat
SIMU 0 5                  "Simulare pe intervalul 0..4
"Se afiseaza rezultatele grafice in doua ferestre pe ecran,
"     una sub cealalta
SPLIT 2 1
ASHOW x1
ASHOW x3
END
```

ANEXA 9.1

Codul sursă al programului *elipsm.pas* este:

```
program elipsoid;
uses crt,printer;
type dimens=record
                 lung:real;
                 nume:string[40]
            end;
    vector=array[1..15] of real;
    kapa=array[1..14] of real;
    lungime=array[1..31] of dimens;
    truncation=array[1..20] of real;

type tc=record
         ColtSs,ColtDs,ColtSj,ColtDj,BaraO,BaraV,BaraV1:char;
         CulFond,CulScris:Byte;
     end;
const
     s:string= '* Constante elastice *';
     s1:string='    MODEL  MECANIC    ';
     contur:tc=(ColtSs:#201;
                ColtDs:#187;
                ColtSj:#200;
                ColtDj:#188;
                BaraO:#205;
                BaraV:#186;
                BaraV1:#186;
                CulFond:0;
                CulScris:15);
var h:integer;
    i1,imr:byte;
    ch1,im:char;
    i,j,t,ch:integer;
    Mtot,Eb,Et,E,tr,suma,Ii,St1:real;
    K,Kt,Kn:kapa;
    L,Lnn:lungime;
    Mi,Min,a,b,c,Si,Sin,an,bn,cn:vector;
    tasta:string;
    cha:char;
    It1,K1raport:truncation;
    n,Mtotn,suma_n,aux:real;

procedure press;
    begin
         tasta:='<< Apasa o tasta pentru a continua >>';
         gotoxy(1,24);
         for i:=1 to (78-length(tasta)) do write(' ');
         write(tasta);
         ch1:=readkey;
     end;

procedure ScrieVert(x,y:byte;sir:string);
   var
      i1:shortint;
   begin
        gotoxy(x,y);
        for i1:=1 to length(sir) do
           begin
                gotoxy(x,y+i1-1);
                write(sir[i1]);
            end;
   end;

procedure chenar(x1,y1,x2,y2:byte);
   var
      i1:byte;
   begin
        with contur do
        begin
             if (x2-x1<=0) or (y2-y1<=0) then Exit;
             textbackground(CulFond);
             textcolor(CulScris);
             gotoxy(x1,y1);
             write(ColtSs);
             for i1:=x1+1 to x2-1 do  write(BaraO);
             write(ColtDs);
             if h=1 then
                  baraV1:=barav
             else baraV1:=#219;
             for i1:=y1+1 to y2-1 do
                 begin
                      gotoxy(x1,i1);
                      write(BaraV1);
                      gotoxy(x2,i1);
                      write(BaraV);
                 end;
             gotoxy(x1,y2);
             write(ColtSj);
             for i1:=x1+1 to x2-1 do write(BaraO);
             write(ColtDj);
        end;
   end;

procedure antet;
 begin
     h:=1;
     textbackground(0);
     clrscr;
```

```
textcolor(15);
scrievert(12,9,'1 9 9 9');
scrievert(69,9,'2 0 0 0');
contur.culscris:=1;
chenar(21,6,60,21);
with contur do
     begin
          coltss:=#176;
          coltds:=#176;
          coltsj:=#176;
          coltdj:=#176;
          barao:=#176;
          barav:=#176;
          culfond:=3;
          culscris:=1;
     end;
chenar(22,7,59,20);
with contur do
     begin
          coltss:='*';
          coltds:='*';
          coltsj:='*';
          coltdj:='*';
          barao:='*';
          barav:='*';
          culfond:=5;
          culscris:=13;
     end;
chenar(23,8,58,19);
with contur do
     begin
          coltss:=#219;
          coltds:=#219;
          coltsj:=#219;
          coltdj:=#219;
          barao:=#219;
          barav:=#186;
          culfond:=5;
          culscris:=14;
     end;
h:=2;
chenar(27,12,53,15);
window(28,13,52,14);
textbackground(red);
textcolor(1);
clrscr;
window(28,13,52,14);
for i1:=1 to 25 do
    begin
         gotoxy(26-i1,1);
         write(copy(s,1,i1));
```

```
              gotoxy(26-1,2);
              write(copy(s1,1,i1));
              delay(200);
         end;
     repeat
           gotoxy(1,1);
           s:=s+s[1];
           delete(s,1,1);
           write(copy(s,1,25));
           gotoxy(1,2);
           s1:=s1+s1[1];
           delete(s1,1,1);
           write(copy(s1,1,25));
           delay(200);
     until keypressed;
end;

procedure vol_n;
  begin
       for i:=1 to 31 do
           case i of
                7,9..16,19..22,24..28,30:
                     begin
                         Lnn[i].lung:=L[i].lung*n;
                         Lnn[i].nume:=L[i].nume;
                     end;
                1..6,8,17,18,23,29,31:  Lnn[i]:=L[i];
           end;
       an[1]:=Lnn[7].lung/2;
       bn[1]:=Lnn[7].lung/2;
       cn[1]:=Lnn[6].lung/2;
       an[2]:=Lnn[9].lung/(2*pi);
       bn[2]:=Lnn[9].lung/(2*pi);
       cn[2]:=(Lnn[1].lung-Lnn[2].lung-Lnn[6].lung)/2;
       an[3]:=Lnn[12].lung/2;
       bn[3]:=Lnn[11].lung/2;
       cn[3]:=Lnn[17].lung/2;
       an[8]:=Lnn[14].lung/2;
       bn[8]:=Lnn[13].lung/2;
       cn[8]:=(Lnn[17].lung+Lnn[18].lung)/4;
       an[9]:=Lnn[16].lung/2;
       bn[9]:=Lnn[15].lung/2;
       cn[9]:=Lnn[18].lung/2;
       an[4]:=Lnn[19].lung/(2*pi);
       bn[4]:=Lnn[19].lung/(2*pi);
       cn[4]:=Lnn[17].lung/2;
       an[5]:=an[4];
       bn[5]:=bn[4];
       cn[5]:=cn[4];
       an[6]:=Lnn[21].lung/(2*pi);
       bn[6]:=Lnn[21].lung/(2*pi);
       cn[6]:=Lnn[18].lung/2;
       an[7]:=an[6];
       bn[7]:=bn[6];
       cn[7]:=cn[6];
       an[10]:=Lnn[25].lung/(2*pi);
       bn[10]:=Lnn[25].lung/(2*pi);
       cn[10]:=(Lnn[2].lung-Lnn[17].lung-Lnn[23].lung)/2;
       an[11]:=an[10];
       bn[11]:=bn[10];
       cn[11]:=cn[10];
       an[12]:=Lnn[27].lung/(2*pi);
       bn[12]:=Lnn[27].lung/(2*pi);
       cn[12]:=(Lnn[23].lung-Lnn[29].lung)/2;
       an[13]:=an[12];
       bn[13]:=bn[12];
       cn[13]:=cn[12];
       an[14]:=Lnn[30].lung/2;
       bn[14]:=Lnn[31].lung/2;
       cn[14]:=Lnn[29].lung/2;
       an[15]:=an[14];
       bn[15]:=bn[14];
       cn[15]:=cn[14];
 end;

procedure listare_dimensiuni;
  begin
     writeln(lst);
     writeln(lst);
     writeln(lst,'    Dimensiuni antropometrice in cazul
m=:',mtot:6:2,' kg');
     writeln(lst,'
================================================');
     for j:=1 to 9 do
         begin
              write(lst,'L ',j,'  ',L[j].nume);
              for i:=1 to (40-length(L[j].nume)) do write(lst,' ');
              writeln(lst,L[j].lung:6:2,' cm');
         end;
     for j:=10 to 20 do
         begin
              write(lst,'L',j,'  ',L[j].nume);
              for i:=1 to (40-length(L[j].nume)) do write(lst,' ');
              writeln(lst,L[j].lung:6:2,' cm');
         end;
     for j:=21 to 31 do
         begin
              write(lst,'L',j,'  ',L[j].nume);
              for i:=1 to (40-length(L[j].nume)) do write(lst,' ');
              writeln(lst,L[j].lung:6:2,' cm');
         end;
     writeln(lst);
     writeln(lst,' Masa totala este m=',mtot:6:2,' kg');
     writeln(lst);
     writeln(lst,' Factorul de trunchiere considerat:
tr=',tr*100:4:2,'%');
end;

procedure listare_intermediar;
   begin
     writeln(lst);
     writeln(lst);
     writeln(lst,'------------------------------------------------
');
     writeln(lst,'  Nr  ai[cm]   bi[cm]   ci[cm]  Mi[kg]  Si[N/cm]
');
     writeln(lst,'------------------------------------------------
');
     for j:=1 to 9 do
         begin
              write(lst,'  ',j,'  ',a[j]:7:3,'  ',b[j]:7:3,'
',c[j]:7:3);
              writeln(lst,'  ',Mi[j]:5:2,'  ',Si[j]:6:1);
         end;
     for j:=10 to 15 do
         begin
              write(lst,' ',j,'  ',a[j]:7:3,'  ',b[j]:7:3,'
',c[j]:7:3);
              writeln(lst,'  ',Mi[j]:5:2,'  ',Si[j]:6:1);
         end;
   end;

procedure listare_k;
  begin
     writeln(lst);
     writeln(lst);
     writeln(lst,'          ---------------');
     writeln(lst,'            Nr    Ki[N/m] ');
     writeln(lst,'          ---------------');
     for j:=1 to 9 do
              writeln(lst,'            K',j,'    ',K[j]:6:1);
     for j:=10 to 14 do
              writeln(lst,'           K',j,'   ',K[j]:7:1);
  end;

procedure listare_k_tr;
  begin
      writeln(lst);
      writeln(lst);
      writeln(lst,'   Variatie K functie de trunchiere:');
      writeln(lst,'   =================================');
      writeln(lst,'          tr[%]   K/K0.05');
      writeln(lst,'          -----   -------');
      for j:=1 to 9 do
          writeln(lst,'            ',j,'     ',K1raport[j]:5:2);
      for j:=10 to 20 do
          writeln(lst,'           ',j,'     ',K1raport[j]:5:2);
  end;

procedure listare_k1_k14_tr;
  begin
     writeln(lst);
     writeln(lst);
     writeln(lst,'           Variatie K functie de trunchiere:');

writeln(lst,'=======================================================
========');
     writeln(lst,'   K1      K2      K3      K4      K5      K6
K7    tr');
     writeln(lst,'--------------------------------------------------
-------------');
     for t:=1 to 9 do
        begin
          It1[t]:=ln((2-t/100)/(t/100));
          for j:=1 to 7 do
            begin
              Kt[j]:=K[j]*Ii/It1[t];
              write(lst,Kt[j]:7:1,' ');
            end;
          writeln(lst,'  ',t);
        end;
     for t:=10 to 20 do
        begin
          It1[t]:=ln((2-t/100)/(t/100));
          for j:=1 to 7 do
            begin
              Kt[j]:=K[j]*Ii/It1[t];
              write(lst,Kt[j]:7:1,' ');
            end;
          writeln(lst,' ',t);
        end;
     writeln(lst);
     writeln(lst,'           Variatie K functie de trunchiere:');

writeln(lst,'=======================================================
========');
     writeln(lst,'   K8      K9     K10     K11     K12     K13
K14    tr');
     writeln(lst,'--------------------------------------------------
-------------');
     for t:=1 to 9 do
        begin
          It1[t]:=ln((2-t/100)/(t/100));
          for j:=8 to 14 do
```

```
            begin
              Kt[j]:=K[j]*Ii/It1[t];
              write(lst,Kt[j]:7:1,' ');
            end;
          writeln(lst,'  ',t);
        end;
     for t:=10 to 20 do
        begin
          It1[t]:=ln((2-t/100)/(t/100));
          for j:=8 to 14 do
            begin
              Kt[j]:=K[j]*Ii/It1[t];
              write(lst,Kt[j]:7:1,' ');
            end;
          writeln(lst,' ',t);
        end;
  end;

procedure listare_m_n;
   begin
          writeln(lst); writeln(lst);
          writeln(lst,'     n[%]    m[kg]');
          writeln(lst,'------------------------');
          n:=0.8;
          repeat
                   vol_n;
                   suma_n:=0;
                   suma:=0;
                   Mtot:=75;
                   for i:=1 to 15 do
                       begin
                            suma:=a[i]*b[i]*c[i]+suma;
                            suma_n:=an[i]*bn[i]*cn[i]+suma_n;
                       end;
                   Mtotn:=Mtot*suma_n/suma;
                   writeln(lst,'    ',n:5:2,'   ',Mtotn:6:2);
                   n:=n+0.02;
          until n>1.2;
   end;

procedure listare_k_n;
   begin
     writeln(lst); writeln(lst);
     writeln(lst,'                   Variatia Kni functie de n');
     writeln(lst,'--------------------------------------------------
---------------------');
     writeln(lst,'    n     Kn1      Kn2      Kn3      Kn4      Kn5
Kn6      Kn7');
     writeln(lst,'--------------------------------------------------
---------------------');
     n:=0.84;
     repeat
           vol_n;
           suma_n:=0;
           suma:=0;
           Mtot:=75;
           for i:=1 to 15 do
               begin
                    suma:=a[i]*b[i]*c[i]+suma;
                    suma_n:=an[i]*bn[i]*cn[i]+suma_n;
               end;
           Mtotn:=Mtot*suma_n/suma;
           Ii:=ln((2-tr)/tr);
           for i:=1 to 15 do
               Mi[i]:=Mtotn*an[i]*bn[i]*cn[i]/suma_n;
           for i:=1 to 15 do
               Si[i]:=pi*E*an[i]*bn[i]*1e-4/(cn[i]*Ii*1e-2);
           K[1]:=Si[1];
           K[2]:=Si[2]*Si[3]/(Si[2]+Si[3]);
           K[3]:=Si[3]*Si[8]/(Si[3]+Si[8]);
           K[4]:=Si[3]*Si[4]/(Si[3]+Si[4]);
           K[5]:=Si[3]*Si[5]/(Si[3]+Si[5]);
           K[6]:=Si[6];
           K[7]:=Si[7];
           for i:=1 to 7 do
               K[i]:=K[i]*100;
           write(lst,' ',n:5:2,' ',K[1]:8:2,' ',K[2]:8:2,'
',K[3]:8:2);
           writeln(lst,' ',K[4]:8:2,' ',K[5]:8:2,' ',K[6]:8:2,'
',K[7]:8:2);
           n:=n+0.04;
     until n>1.2;
     writeln(lst,'--------------------------------------------------
-----------------');
     writeln(lst,'    n     Kn8      Kn9     Kn10     Kn11     Kn12
Kn13     Kn14');
     writeln(lst,'--------------------------------------------------
-----------------');
     n:=0.84;
     repeat
           vol_n;
           suma_n:=0;
           suma:=0;
           for i:=1 to 15 do
               begin
                    suma:=a[i]*b[i]*c[i]+suma;
                    suma_n:=an[i]*bn[i]*cn[i]+suma_n;
               end;
           Mtotn:=Mtot*suma_n/suma;
           for i:=1 to 15 do
               Mi[i]:=Mtotn*an[i]*bn[i]*cn[i]/suma_n;
           for i:=1 to 15 do
```

```
               Si[i]:=pi*E*an[i]*bn[i]*1e-4/(cn[i]*Ii*1e-2);
           K[8]:=Si[8]*Si[9]/(Si[8]+Si[9]);
           K[9]:=Si[9]*Si[10]/(Si[9]+Si[10]);
           K[10]:=Si[9]*Si[11]/(Si[9]+Si[11]);
           K[11]:=Si[10]*Si[12]/(Si[10]+Si[12]);
           K[12]:=Si[11]*Si[13]/(Si[11]+Si[13]);
           K[13]:=Si[14];
           K[14]:=Si[15];
           for i:=8 to 14 do
               K[i]:=K[i]*100;
           write(lst,' ',n:5:2,' ',K[8]:8:2,' ',K[9]:8:2,'
',K[10]:8:2);
           writeln(lst,' ',K[11]:8:2,' ',K[12]:8:2,' ',K[13]:8:2,'
',K[14]:8:2);
           n:=n+0.04;
     until n>1.2;
   end;

procedure imprimanta;
   begin
     window(25,22,78,24);
     gotoxy(12,1);
     textbackground(7);
     textcolor(10);
     writeln;
     writeln('Doresti sa tiparesti la imprimanta aceste valori?');
     write('    [d/n]         ');
     readln(im);
     if (im='d') or (im='D') then
        case imr of
          1: listare_dimensiuni;
          2: listare_intermediar;
          3: listare_k;
          4: listare_k_tr;
          5: listare_k1_k14_tr;
          6: listare_m_n;
          7: listare_k_n;
        end;
     window(1,1,79,25);
     textbackground(3);
     textcolor(15);
     clrscr;
  end;

begin
     clrscr;
     antet;
     window(1,1,79,25);
     textbackground(3);
     textcolor(15);
     clrscr;
```

```
gotoxy(10,10);
writeln('Alege:');
writeln('   1.  Date implicite');
writeln('   2.  Date introduse de la tastatura');
readln(ch);
Eb:=22.6e9;
Et:=7.5e3;
E:=sqrt(Eb*Et);  {N/m2}
E:=E/10000;      {N/cm2}
L[1].nume:='Inaltime totala';
L[2].nume:='Inaltimea pana la umeri';
L[3].nume:='Inaltimea pana la incheietura mainii';
L[4].nume:='Inaltimea pana la talie';
L[5].nume:='Inaltimea pana la sezut';
L[6].nume:='Lungimea capului';
L[7].nume:='Latimea capului';
L[8].nume:='Distanta de la cap la barbie';
L[9].nume:='Circumferinta gatului';
L[10].nume:='Latimea umarului';
L[11].nume:='Grosimea pieptului';
L[12].nume:='Latimea pieptului';
L[13].nume:='Grosimea taliei';
L[14].nume:='Latimea taliei';
L[15].nume:='Grosime fese';
L[16].nume:='Latime sold in picioare';
L[17].nume:='Distanta dintre umar si cot';
L[18].nume:='Lungimea antebratului';
L[19].nume:='Circumferinta bicepsului';
L[20].nume:='Circumferinta cotului';
L[21].nume:='Circumferinta antebratului';
L[22].nume:='Circumferinta incheieturii mainii';
L[23].nume:='Inaltimea pana la genunchi';
L[24].nume:='Circumferinta coapsei (pulpei)';
L[25].nume:='Circumferinta piciorului superior';
L[26].nume:='Circumferinta genunchiului';
L[27].nume:='Circumferinta gambei';
L[28].nume:='Circumferinta gleznei';
L[29].nume:='Inaltimea gambei';
L[30].nume:='Latimea labei piciorului';
L[31].nume:='Lungimea labei piciorului';
if ch=1 then
   begin
        Mtot:=70;
        L[1].lung:=171;
        L[2].lung:=146;
        L[3].lung:=133;
        L[4].lung:=105;
        L[5].lung:=93;
        L[6].lung:=21;
```

```
             L[7].lung:=17;
             L[8].lung:=24;
             L[9].lung:=38;
             L[10].lung:=46;
             L[11].lung:=24;
             L[12].lung:=35;
             L[13].lung:=27;
             L[14].lung:=28;
             L[15].lung:=22;
             L[16].lung:=35;
             L[17].lung:=39;
             L[18].lung:=49;
             L[19].lung:=35;
             L[20].lung:=29;
             L[21].lung:=27;
             L[22].lung:=17;
             L[23].lung:=51;
             L[24].lung:=48;
             L[25].lung:=40;
             L[26].lung:=38;
             L[27].lung:=36;
             L[28].lung:=22;
             L[29].lung:=7;
             L[30].lung:=10;
             L[31].lung:=26;
             tr:=0.05;
         end
     else
         begin
              clrscr;
              writeln;
              write('Dati masa totala a corpului uman!    m=');
              readln(Mtot);
              writeln('Dati urmatoarele dimensiuni in cm:');
              for i:=1 to 31 do
                  begin
                       write('    ',L[i].nume,': ');
                       readln(L[i].lung);
                  end;
              writeln;
              writeln('Dati valoarea factorului de trunchiere in
%!');
              write('   (obisnuit este 5%)       tr=');
              readln(tr);
              tr:=tr/100;
        end;
     a[1]:=L[7].lung/2;
     b[1]:=L[7].lung/2;
     c[1]:=L[6].lung/2;
     a[2]:=L[9].lung/(2*pi);
     b[2]:=L[9].lung/(2*pi);
```

```
c[2]:=(L[1].lung-L[2].lung-L[6].lung)/2;
a[3]:=L[12].lung/2;
b[3]:=L[11].lung/2;
c[3]:=L[17].lung/2;
a[8]:=L[14].lung/2;
b[8]:=L[13].lung/2;
c[8]:=(L[17].lung+L[18].lung)/4;
a[9]:=L[16].lung/2;
b[9]:=L[15].lung/2;
c[9]:=L[18].lung/2;
a[4]:=L[19].lung/(2*pi);
b[4]:=L[19].lung/(2*pi);
c[4]:=L[17].lung/2;
a[5]:=a[4];
b[5]:=b[4];
c[5]:=c[4];
a[6]:=L[21].lung/(2*pi);
b[6]:=L[21].lung/(2*pi);
c[6]:=L[18].lung/2;
a[7]:=a[6];
b[7]:=b[6];
c[7]:=c[6];
a[10]:=L[25].lung/(2*pi);
b[10]:=L[25].lung/(2*pi);
c[10]:=(L[2].lung-L[17].lung-L[23].lung)/2;
a[11]:=a[10];
b[11]:=b[10];
c[11]:=c[10];
a[12]:=L[27].lung/(2*pi);
b[12]:=L[27].lung/(2*pi);
c[12]:=(L[23].lung-L[29].lung)/2;
a[13]:=a[12];
b[13]:=b[12];
c[13]:=c[12];
a[14]:=L[30].lung/2;
b[14]:=L[31].lung/2;
c[14]:=L[29].lung/2;
a[15]:=a[14];
b[15]:=b[14];
c[15]:=c[14];
Ii:=ln((2-tr)/tr);
suma:=0;
for i:=1 to 15 do
    suma:=a[i]*b[i]*c[i]+suma;
for i:=1 to 15 do
         Mi[i]:=Mtot*a[i]*b[i]*c[i]/suma;
for i:=1 to 15 do
    Si[i]:=pi*E*a[i]*b[i]*1e-4/(c[i]*Ii*1e-2);
K[1]:=Si[1];
K[2]:=Si[2]*Si[3]/(Si[2]+Si[3]);
K[3]:=Si[3]*Si[8]/(Si[3]+Si[8]);
```

```
     K[4]:=Si[3]*Si[4]/(Si[3]+Si[4]);
     K[5]:=Si[3]*Si[5]/(Si[3]+Si[5]);
     K[6]:=Si[6];
     K[7]:=Si[7];
     K[8]:=Si[8]*Si[9]/(Si[8]+Si[9]);
     K[9]:=Si[9]*Si[10]/(Si[9]+Si[10]);
     K[10]:=Si[9]*Si[11]/(Si[9]+Si[11]);
     K[11]:=Si[10]*Si[12]/(Si[10]+Si[12]);
     K[12]:=Si[11]*Si[13]/(Si[11]+Si[13]);
     K[13]:=Si[14];
     K[14]:=Si[15];
     clrscr;
     writeln;
     writeln('      Dimensiuni antropometrice in cazul
m=',mtot:6:2,' kg');
     writeln('
==============================================');
     for j:=1 to 9 do
         begin
              write('L ',j,'  ',L[j].nume);
              for i:=1 to (40-length(L[j].nume)) do write(' ');
              writeln(L[j].lung:6:2,' cm');
         end;
     for j:=10 to 20 do
         begin
              write('L',j,'  ',L[j].nume);
              for i:=1 to (40-length(L[j].nume)) do write(' ');
              writeln(L[j].lung:6:2,' cm');
         end;
     press;
     clrscr;
     for j:=21 to 31 do
         begin
              write('L',j,'  ',L[j].nume);
              for i:=1 to (40-length(L[j].nume)) do write(' ');
              writeln(L[j].lung:6:2,' cm');
         end;
     writeln;writeln;
     writeln(' Masa totala este:  m=',mtot:6:2,' kg');
     writeln;writeln;
     writeln(' Factorul de trunchiere considerat:
tr=',tr*100:4:2,'%');
     imr:=1;
     delay(300);
     imprimanta;
     writeln;
     writeln('------------------------------------------------');
     writeln('  Nr  ai[cm]   bi[cm]   ci[cm]  Mi[kg]  Si[N/cm] ');
     writeln('------------------------------------------------');
     for j:=1 to 9 do
         begin
```

```
              write('  ',j,'  ',a[j]:7:3,'  ',b[j]:7:3,'
',c[j]:7:3);
              writeln('  ',Mi[j]:5:2,'  ',Si[j]:6:1);
         end;
     for j:=10 to 15 do
         begin
              write(' ',j,'  ',a[j]:7:3,'  ',b[j]:7:3,'
',c[j]:7:3);
              writeln('  ',Mi[j]:5:2,'  ',Si[j]:6:1);
         end;
     imr:=2;
     delay(300);
     imprimanta;
     for j:=1 to 14 do
         K[j]:=K[j]*100;       {N/m}
     writeln;
     writeln('          ---------------');
     writeln('            Nr    Ki[N/m] ');
     writeln('          ---------------');
     for j:=1 to 9 do
              writeln('            K',j,'    ',K[j]:6:1);
     for j:=10 to 14 do
              writeln('           K',j,'   ',K[j]:7:1);
     imr:=3;
     delay(300);
     imprimanta;
     writeln; writeln; writeln; writeln; writeln;
     writeln('  Doriti  sa continuati cu un studiu de analiza ');
     writeln('  a variatiei parametrilor ?');
     write('     [D/N]      ');
     readln(cha);
     if (cha='d') or (cha='D') then
        begin
             for t:=1 to 20 do
                begin
                     It1[t]:=ln((2-t/100)/(t/100));
                     K1raport[t]:=Ii/It1[t];
                end;
             clrscr;
             writeln('   Variatie K functie de trunchiere:');
             writeln('   =================================');
             writeln('          tr[%]   K/K0.05');
             writeln('          -----   -------');
             for j:=1 to 9 do
                 writeln('           ',j,'     ',K1raport[j]:5:2);
             for j:=10 to 20 do
                 writeln('           ',j,'     ',K1raport[j]:5:2);
             imr:=4;
             delay(300);
             imprimanta;
             clrscr;
     writeln('           Variatie K functie de trunchiere:');

writeln('=========================================================
======');
     writeln('   K1      K2      K3      K4      K5      K6
K7    tr');
     writeln('----------------------------------------------------
-----------');
     for t:=1 to 9 do
        begin
          It1[t]:=ln((2-t/100)/(t/100));
          for j:=1 to 7 do
            begin
              Kt[j]:=K[j]*Ii/It1[t];
              write(Kt[j]:7:1,' ');
            end;
          writeln('  ',t);
        end;
     for t:=10 to 20 do
        begin
          It1[t]:=ln((2-t/100)/(t/100));
          for j:=1 to 7 do
            begin
              Kt[j]:=K[j]*Ii/It1[t];
              write(Kt[j]:7:1,' ');
            end;
          writeln(' ',t);
        end;
     press;
     clrscr;
     writeln('           Variatie K functie de trunchiere:');

writeln('=========================================================
======');
     writeln('   K8      K9     K10     K11     K12     K13
K14    tr');
     writeln('----------------------------------------------------
-----------');
     for t:=1 to 9 do
        begin
          It1[t]:=ln((2-t/100)/(t/100));
          for j:=8 to 14 do
            begin
              Kt[j]:=K[j]*Ii/It1[t];
              write(Kt[j]:7:1,' ');
            end;
          writeln('  ',t);
        end;
     for t:=10 to 20 do
        begin
          It1[t]:=ln((2-t/100)/(t/100));
```

```
       for j:=8 to 14 do
         begin
           Kt[j]:=K[j]*Ii/It1[t];
           write(Kt[j]:7:1,' ');
         end;
       writeln(' ',t);
     end;
   imr:=5;
   delay(700);
   imprimanta;
   writeln; writeln; writeln; writeln; writeln;
   writeln('    Continuati studiul pentru a analiza variatia');
   writeln('parametrilor modelului mecanic functie de latime');
   writeln('pentru persoane care au aceeasi inaltime ? ');
   write('          [D/N]    ');
   readln(cha);
   if (cha='d') or (cha='D') then
      begin
        tr:=0.05;
        clrscr;
        writeln;
        writeln('    n[-]   m[kg]');
        writeln('----------------------');
        n:=0.8;
        repeat
             vol_n;
             suma_n:=0;
             suma:=0;
             Mtot:=75;
             for i:=1 to 15 do
                 begin
                      suma:=a[i]*b[i]*c[i]+suma;
                      suma_n:=an[i]*bn[i]*cn[i]+suma_n;
                 end;
             Mtotn:=Mtot*suma_n/suma;
             writeln('    ',n:5:2,'   ',Mtotn:6:2);
             n:=n+0.02;
        until n>1.2;
          imr:=6;
          delay(350);
          imprimanta;
          writeln;
          writeln('    n     Kn1      Kn2      Kn3      Kn4
Kn5      Kn6      Kn7');
          writeln('----------------------------------------------
--------------------------');
          n:=0.84;
          repeat
             vol_n;
             suma_n:=0;
             suma:=0;
```

```
                   Mtot:=75;
                   for i:=1 to 15 do
                       begin
                            suma:=a[i]*b[i]*c[i]+suma;
                            suma_n:=an[i]*bn[i]*cn[i]+suma_n;
                       end;
                   Mtotn:=Mtot*suma_n/suma;
                   Ii:=ln((2-tr)/tr);
                   for i:=1 to 15 do
                       Mi[i]:=Mtotn*an[i]*bn[i]*cn[i]/suma_n;
                   for i:=1 to 15 do
                       Si[i]:=pi*E*an[i]*bn[i]*1e-4/(cn[i]*Ii*1e-
2);
                   K[1]:=Si[1];
                   K[2]:=Si[2]*Si[3]/(Si[2]+Si[3]);
                   K[3]:=Si[3]*Si[8]/(Si[3]+Si[8]);
                   K[4]:=Si[3]*Si[4]/(Si[3]+Si[4]);
                   K[5]:=Si[3]*Si[5]/(Si[3]+Si[5]);
                   K[6]:=Si[6];
                   K[7]:=Si[7];
                   for i:=1 to 7 do
                       K[i]:=K[i]*100;
                   write(' ',n:5:2,' ',K[1]:8:2,' ',K[2]:8:2,'
',K[3]:8:2);
                   writeln(' ',K[4]:8:2,' ',K[5]:8:2,'
',K[6]:8:2,' ',K[7]:8:2);
                   n:=n+0.04;
               until n>1.2;
               delay(450);
               writeln('------------------------------------------
-------------------------');
               writeln('    n     Kn8      Kn9     Kn10     Kn11
Kn12     Kn13     Kn14');
               writeln('------------------------------------------
-------------------------');
               n:=0.84;
               repeat
                   vol_n;
                   suma_n:=0;
                   suma:=0;
                   for i:=1 to 15 do
                       begin
                            suma:=a[i]*b[i]*c[i]+suma;
                            suma_n:=an[i]*bn[i]*cn[i]+suma_n;
                       end;
                   Mtotn:=Mtot*suma_n/suma;
                   for i:=1 to 15 do
                       Mi[i]:=Mtotn*an[i]*bn[i]*cn[i]/suma_n;
                   for i:=1 to 15 do
                   Si[i]:=pi*E*an[i]*bn[i]*1e-4/(cn[i]*Ii*1e-2);
                   K[8]:=Si[8]*Si[9]/(Si[8]+Si[9]);
                   K[9]:=Si[9]*Si[10]/(Si[9]+Si[10]);
                   K[10]:=Si[9]*Si[11]/(Si[9]+Si[11]);
                   K[11]:=Si[10]*Si[12]/(Si[10]+Si[12]);
                   K[12]:=Si[11]*Si[13]/(Si[11]+Si[13]);
                   K[13]:=Si[14];
                   K[14]:=Si[15];
                   for i:=8 to 14 do
                       K[i]:=K[i]*100;
                   write(' ',n:5:2,' ',K[8]:8:2,' ',K[9]:8:2,'
',K[10]:8:2);
                   writeln(' ',K[11]:8:2,' ',K[12]:8:2,'
',K[13]:8:2,' ',K[14]:8:2);
                   n:=n+0.04;
               until n>1.2;
               imr:=7;
               delay(350);
               imprimanta;
          end;
        end;
     clrscr;
     gotoxy(10,12);
     textbackground(3);
     textcolor(11);
     writeln('VA MULTUMIM PENTRU CA ATI FOLOSIT ACEST PROGRAM !');
     gotoxy(5,18);
     writeln('Profesor coordonator:');
     writeln(' prof.univ.dr.ing.Liviu BRINDEU');
     gotoxy(60,18);
     writeln('Doctorand:');
     gotoxy(58,19);
     writeln('ing.Cornel POPA');
     delay(6500);
end.
```

ANEXA 9.2

Listingul obținut prin rularea programului *elipsm.exe*:

```
    Dimensiuni antropometrice in cazul m=: 70.00 kg
    ===============================================
L 1  Inaltime totala                            171.00 cm
L 2  Inaltimea pana la umeri                    146.00 cm
L 3  Inaltimea pana la incheietura mainii       133.00 cm
L 4  Inaltimea pana la talie                    105.00 cm
L 5  Inaltimea pana la sezut                     93.00 cm
L 6  Lungimea capului                            21.00 cm
L 7  Latimea capului                             17.00 cm
L 8  Distanta de la cap la barbie                24.00 cm
L 9  Circumferinta gatului                       38.00 cm
L10  Latimea umarului                            46.00 cm
L11  Grosimea pieptului                          24.00 cm
L12  Latimea pieptului                           35.00 cm
L13  Grosimea taliei                             27.00 cm
L14  Latimea taliei                              28.00 cm
L15  Grosime fese                                22.00 cm
L16  Latime sold in picioare                     35.00 cm
L17  Distanta dintre umar si cot                 39.00 cm
L18  Lungimea antebratului                       49.00 cm
L19  Circumferinta bicepsului                    35.00 cm
L20  Circumferinta cotului                       29.00 cm
L21  Circumferinta antebratului                  27.00 cm
L22  Circumferinta incheieturii mainii           17.00 cm
L23  Inaltimea pana la genunchi                  51.00 cm
L24  Circumferinta coapsei (pulpei)              48.00 cm
L25  Circumferinta piciorului superior           40.00 cm
L26  Circumferinta genunchiului                  38.00 cm
L27  Circumferinta gambei                        36.00 cm
L28  Circumferinta gleznei                       22.00 cm
L29  Inaltimea gambei                             7.00 cm
L30  Latimea labei piciorului                    10.00 cm
L31  Lungimea labei piciorului                   26.00 cm

 Masa totala este m= 70.00 kg

 Factorul de trunchiere considerat:     tr=5.00%


-----------------------------------------------
  Nr  ai[cm]   bi[cm]   ci[cm]  Mi[kg]  Si[N/cm]
-----------------------------------------------
  1    8.500    8.500   10.500   2.94    76.8
  2    6.048    6.048    2.000   0.28   204.2
  3   17.500   12.000    9.750   7.95   240.5
  4    5.570    5.570   19.500   2.35    17.8
  5    5.570    5.570   19.500   2.35    17.8
  6    4.297    4.297   24.500   1.76     8.4
  7    4.297    4.297   24.500   1.76     8.4
  8   14.000   13.500   22.000  16.14    95.9
  9   17.500   11.000   24.500  18.30    87.7
 10    6.366    6.366   28.000   4.40    16.2
 11    6.366    6.366   28.000   4.40    16.2
 12    5.730    5.730   22.000   2.80    16.7
 13    5.730    5.730   22.000   2.80    16.7
 14    5.000   13.000    3.500   0.88   207.3
 15    5.000   13.000    3.500   0.88   207.3
```

Nr	Ki[N/m]
K1	7682.1
K2	11042.0
K3	6856.4
K4	1654.3
K5	1654.3
K6	841.5
K7	841.5
K8	4581.6
K9	1364.6
K10	1364.6
K11	820.3
K12	820.3
K13	20733.7
K14	20733.7

Variatie K functie de trunchiere:

tr[%]	K/K0.05
1	0.69
2	0.80
3	0.88
4	0.94
5	1.00
6	1.05
7	1.10
8	1.15
9	1.20
10	1.24
11	1.29
12	1.33
13	1.37
14	1.42
15	1.46
16	1.50
17	1.54
18	1.58
19	1.63
20	1.67

Variatie K functie de trunchiere:

K1	K2	K3	K4	K5	K6	K7	tr
5316.9	7642.3	4745.4	1145.0	1145.0	582.4	582.4	1
6124.7	8803.5	5466.4	1318.9	1318.9	670.9	670.9	2
6725.6	9667.1	6002.7	1448.3	1448.3	736.7	736.7	3
7231.5	10394.3	6454.2	1557.3	1557.3	792.1	792.1	4
7682.1	11042.0	6856.4	1654.3	1654.3	841.5	841.5	5
8096.4	11637.5	7226.1	1743.5	1743.5	886.8	886.8	6
8485.3	12196.4	7573.2	1827.3	1827.3	929.4	929.4	7
8855.7	12728.8	7903.8	1907.0	1907.0	970.0	970.0	8
9212.2	13241.3	8222.1	1983.8	1983.8	1009.1	1009.1	9
9558.3	13738.8	8530.9	2058.3	2058.3	1047.0	1047.0	10
9896.4	14224.7	8832.7	2131.1	2131.1	1084.0	1084.0	11
10228.4	14701.9	9129.0	2202.7	2202.7	1120.4	1120.4	12
10555.9	15172.7	9421.3	2273.2	2273.2	1156.2	1156.2	13
10880.3	15638.9	9710.8	2343.0	2343.0	1191.8	1191.8	14
11202.4	16101.9	9998.3	2412.4	2412.4	1227.1	1227.1	15
11523.3	16563.1	10284.7	2481.5	2481.5	1262.2	1262.2	16
11843.7	17023.7	10570.7	2550.5	2550.5	1297.3	1297.3	17
12164.3	17484.6	10856.8	2619.5	2619.5	1332.4	1332.4	18

```
12485.9 17946.7 11143.8  2688.8  2688.8  1367.6  1367.6  19
12808.8 18410.9 11432.0  2758.3  2758.3  1403.0  1403.0  20

           Variatie K functie de trunchiere:
===============================================================
   K8      K9      K10      K11      K12      K13      K14    tr
---------------------------------------------------------------
 3171.0   944.4   944.4   567.7    567.7 14350.0 14350.0   1
 3652.8  1087.9  1087.9   654.0    654.0 16530.4 16530.4   2
 4011.2  1194.7  1194.7   718.1    718.1 18152.1 18152.1   3
 4312.9  1284.6  1284.6   772.2    772.2 19517.6 19517.6   4
 4581.6  1364.6  1364.6   820.3    820.3 20733.7 20733.7   5
 4828.7  1438.2  1438.2   864.5    864.5 21851.8 21851.8   6
 5060.7  1507.3  1507.3   906.0    906.0 22901.5 22901.5   7
 5281.6  1573.1  1573.1   945.6    945.6 23901.1 23901.1   8
 5494.2  1636.4  1636.4   983.7    983.7 24863.5 24863.5   9
 5700.6  1697.9  1697.9  1020.6   1020.6 25797.5 25797.5  10
 5902.2  1757.9  1757.9  1056.7   1056.7 26710.0 26710.0  11
 6100.3  1816.9  1816.9  1092.2   1092.2 27606.1 27606.1  12
 6295.6  1875.1  1875.1  1127.1   1127.1 28490.1 28490.1  13
 6489.0  1932.7  1932.7  1161.8   1161.8 29365.4 29365.4  14
 6681.2  1989.9  1989.9  1196.2   1196.2 30234.8 30234.8  15
 6872.5  2046.9  2046.9  1230.4   1230.4 31100.9 31100.9  16
 7063.6  2103.8  2103.8  1264.7   1264.7 31965.7 31965.7  17
 7254.9  2160.8  2160.8  1298.9   1298.9 32831.1 32831.1  18
 7446.6  2217.9  2217.9  1333.2   1333.2 33698.8 33698.8  19
 7639.2  2275.3  2275.3  1367.7   1367.7 34570.5 34570.5  20


     n[%]    m[kg]
----------------------
     0.80    48.30
     0.82    50.71
     0.84    53.17
     0.86    55.70
     0.88    58.28
     0.90    60.92
     0.92    63.62
     0.94    66.38
     0.96    69.19
     0.98    72.07
     1.00    75.00
     1.02    77.99
     1.04    81.04
     1.06    84.15
     1.08    87.32
     1.10    90.54
     1.12    93.83
     1.14    97.17
     1.16   100.57
     1.18   104.03
```

Variatia Kni functie de n

n	Kn1	Kn2	Kn3	Kn4	Kn5	Kn6	Kn7
0.84	5420.49	7791.21	4837.86	1167.28	1167.28	593.73	593.73
0.88	5949.02	8550.90	5309.58	1281.10	1281.10	651.63	651.63
0.92	6502.13	9345.92	5803.24	1400.21	1400.21	712.21	712.21
0.96	7079.82	10176.28	6318.84	1524.61	1524.61	775.49	775.49
1.00	7682.10	11041.97	6856.38	1654.31	1654.31	841.46	841.46
1.04	8308.96	11942.99	7415.86	1789.30	1789.30	910.12	910.12
1.08	8960.40	12879.35	7997.28	1929.59	1929.59	981.48	981.48
1.12	9636.42	13851.04	8600.64	2075.17	2075.17	1055.53	1055.53
1.16	10337.03	14858.07	9225.95	2226.04	2226.04	1132.27	1132.27

n	Kn8	Kn9	Kn10	Kn11	Kn12	Kn13	Kn14
0.84	3232.80	962.85	962.85	578.79	578.79	17416.30	17416.30
0.88	3548.02	1056.74	1056.74	635.23	635.23	18245.64	18245.64
0.92	3877.90	1154.99	1154.99	694.29	694.29	19074.99	19074.99
0.96	4222.44	1257.60	1257.60	755.97	755.97	19904.34	19904.34
1.00	4581.64	1364.59	1364.59	820.28	820.28	20733.69	20733.69
1.04	4955.50	1475.94	1475.94	887.22	887.22	21563.03	21563.03
1.08	5344.02	1591.65	1591.65	956.78	956.78	22392.38	22392.38
1.12	5747.20	1711.74	1711.74	1028.96	1028.96	23221.73	23221.73
1.16	6165.05	1836.19	1836.19	1103.77	1103.77	24051.08	24051.08

CUPRINS