



```

PROGRAM DECKEL;
{napisal Waclaw Libront 1990}
uses tpcrt,dos,lw_okna;
type wartosci=(x,y,z,c,n,t);
    ktore_meni=(pomocy,procedury);
    typy_list=(wewy,bloki,funkcje);
    typy_koriobr=(wlanzczamy,wylanczamy);
    oddo=0..999;
const czasy:array[1..20] of real=
    (12.5,16,20,25,31.5,40,50,63,80,100,125,150,200,250,315,400,500,630,800,1000);
    tablica:array [16..50] of char='qwertyuiop****asdfghjkl*****zxcvbnm';
    tryb:byte=23;
    ile_wierszy:byte=20;
    tkorekcji:real=20;
    tpomiaru:real=60;
    tobrotow:real=10;
    tzmianynarz:real=300;
    sb:string[63]='\pt\dec\';
    sd:string[63]='\pt\dec\';
    rb:string[10]='.OSN';
    rd:string[10]='`.OSN';
    ekrt:byte=0;
    ekr:byte=7;
    ekrl:byte=15;
    podt:byte=1;
    podl:byte=14;
    podr:byte=2;
    pods:byte=0;
    podn:byte=14;
    podm:byte=4;
    poziom_zapisu:oddo=50;
    razem_z_dziobem:boolean=true;
var pwb,nw:byte;
    ile,kop,nb_max,nb,nb_max1:0..999;
    gl:y..z;
    xbaz,ybaz,zbaz,cbaz,nbaz:real;
    kb,fb:char;
    pb,ob:string[2];
    wb:string[5];
    ol,pl,kl,fl:byte;
    wl:real;
    tabb:array[0..999,1..16] of char;
    tabl:array[0..999,x..t] of real;
    f,l:array[1..80] of char;
    x1,x2,y1,y2:byte;
    obraca_sie:boolean;
    zapis_kontrolny:boolean;
    poziom:integer;

procedure zmiana_narzedzia; forward;
procedure save; forward;

```

```

procedure podswietl (x,y,long,high:byte);
var h,i,j,k,lt,at:byte;
begin
  for h:=0 to (high-1) do
    for i:=0 to (long-1) do
      begin getchar(x+i,y+h,lt,at); gotoxy(x+i,y+h);write(char(lt)); end
    end;
end;

function fileexist(nazwa:string):boolean;
var plik:file;
    istnieje:boolean;
begin assign(plik,nazwa);
      {$I-}
      reset(plik);
      {$I+}
      istnieje:=(ioresult=0);
      if istnieje then close(plik);
      fileexist:=istnieje;
end;

function lancuch(dlug:byte):string; {odczytanie z klawiatury ciagu znakow}
var znak:char; x,y,poz:byte;
    bufor:array[1..80] of char;

  procedure wprawo;
    begin if poz<dlug then poz:=poz+1 end;
  procedure wlewo;
    begin if poz>1 then poz:=poz-1 end;

begin
  poz:=1;for x:=1 to dlug do bufor[x]:=#32;
  x:=wherex;y:=wherey;
  repeat
    while keypressed do znak:=readkey;
      gotoxy(x+poz,y);{write('_')}
      znak:=readkey;
      case znak of
        #27:lancuch:=#32;
        #32..#255:begin
          znak:=upcase(znak);
          bufor[poz]:=znak;
          gotoxy(x+poz,y);write(znak);
          wprawo
          end;
        #13:lancuch:=copy(bufor,1,poz-1);
        #8:begin
          wlewo;
          bufor[poz]:=#32;
          gotoxy(x+poz,y);write(' ')
          end
      end;
    until (znak=#13) or (znak=#27)
  end;

procedure beep(ile:byte); {sygnal dzwiekowy}
var j:byte;
begin
  for j:=1 to ile do
    begin
      sound(300);delay(100);nosound;delay(100)
    end
  end;
end;

```

```

procedure intread(var p:integer); {bezpieczne wczytanie liczby integer}
var ok:boolean;x,y:byte;liczba:string;kod:integer;
begin
  x:=wherex-2;y:=wherey;
  repeat
    gotoxy(x,y);write('          ');gotoxy(x,y);
    liczba:=lancuch(4);
    val(liczba,p,kod);
    if kod<>0 then beep(2)
  until kod=0
end;

procedure realread(var p:real); {bezpieczne wczytanie liczby real}
var ok:boolean;x,y:byte;liczba:string;kod:integer;
begin
  x:=wherex-2;y:=wherey;
  repeat
    gotoxy(x,y);write('          ');gotoxy(x,y);
    liczba:=lancuch(9);
    val(liczba,p,kod);
    if kod<>0 then beep(2)
  until kod=0
end;

procedure kolor(tlo,znak:byte);
begin textbackground(tlo);textcolor(znak) end;

procedure do_linii (co:wartosci;numb:oddo);
var od,i:byte;
    c:string[8];
begin
  od:=23+ord(co)*8;str(tabl[numb,co]:8:2,c);
  for i:=1 to 8 do l[i+od]:=c[i]
end;

procedure cala_linia (nbn:oddo);
var w:wartosci;
begin for w:=x to t do do_linii (w,nbn) end;

procedure wysw_linie (nwn,jak:byte);
var i,nw1:byte;d:string[8];
    nb1:0..999;
begin
  l:=f;nw1:=tryb-ile_wierszy+nwn;nb1:=nb-nw+nwn;
  gotoxy(1,nw1);for i:= 1 to 16 do l[i]:=tabb[nb1,i];
  if (tabb[nb1,16]='1') and (tabb[nb1,13]<>'9') then l[19]:='R';
  if jak=1 then
    begin
      cala_linia(nb1);
      gotoxy(wherex,63);
      str((tabl[nb1,t]-tabl[(nb1-1),t]):8:2,d);
      for i:=1 to 8 do l[i+63]:=d[i];
      kolor(podr,pods)
    end
  else
    begin
      case fl of
        9:cala_linia (nb1);
        1,3:begin
          case kl of
            1,2:do_linii (x,nb1);
            3,4:do_linii (y,nb1);
            5,6:do_linii (z,nb1);
            7,8:do_linii (c,nb1)
          end;
          do_linii (t,nb1)
        end;
      end;
    end;
end;

```

```

        end;
        7,8:do_linii(t,nb1)
        end;
        kolor(ekrt,ekrl)
        end;
        write(l)
    end;
end;

procedure na_segmenty;
    var w:array[5..16] of byte; i:byte; j:integer;
begin
    for i:=5 to 16 do
        if tabb[nb,i] in [#48..#57] then val(tabb[nb,i],w[i],j);
        kl:=w[5];kb:=tabb[nb,5];
        wl:=w[7]*100+w[8]*10+w[9]+w[10]*0.1+w[11]*0.01;
        if (kl=7) or (kl=8) then wl:=wl/10;
        wb:=tabb[nb,7]+tabb[nb,8]+tabb[nb,9]+tabb[nb,10]+tabb[nb,11];
        pl:=w[13]*10+w[14];
        fl:=w[16];fb:=tabb[nb,16];
        if (kl=2) or (kl=4) or (kl=6) or (kl=8) then wl:=-wl
    end;

function czas:real;
    var cz:real;
begin
    na_segmenty;
    if ((pl=99) or (pl=0)) then cz:=500 else cz:=czas[pl];
    if ((fl=1) or (fl=3))
        then
            if wl<>0 then cz:= (wl/cz)*60
                else cz:= tkorekcji;
    case fl of
        7:cz:=tobrotow;
        8:cz:=tpomiaru;
        9:cz:=tzmianynarz
    end;
    czas:=abs(cz/60)
end;

procedure w_gore; forward;

procedure nowe_wsp;
    var w:wartosci;
begin
    case fl of
        9:begin
            if tabl[nb,n]=0 then begin zmiana_narzedzia; w_gore end;
            tabl[nb,gl]:=(tabl[nb-1,gl]+tabl[nb-1,n])-tabl[nb,n];
            tabl[nb,x]:=tabl[nb-1,x];
            if gl=y then tabl[nb,z]:=tabl[nb-1,z]
                else tabl[nb,y]:=tabl[nb-1,y];
            tabl[nb,c]:=tabl[nb-1,c];
        end;
        2,5,7,8:for w:=x to n do tabl[nb,w]:=tabl[nb-1,w];
        1,3:begin
            for w:=x to n do tabl[nb,w]:=tabl[nb-1,w];
            case kl of
                1,2: tabl[nb,x]:=tabl[nb-1,x]+wl;
                3,4: tabl[nb,y]:=tabl[nb-1,y]+wl;
                5,6: tabl[nb,z]:=tabl[nb-1,z]+wl;
                7,8: begin
                    tabl[nb,c]:=tabl[nb-1,c]+wl;
                    if abs(tabl[nb,c])=360 then tabl[nb,c]:=0
                end
            end
        end;
    end;
end;
end;
end;

```

```

    tabl[nb,t]:=tabl[nb-1,t]+czas
end;

procedure w_prawo;
    begin if pwb<16 then pwb:=pwb+1 else pwb:=5 end;

procedure w_lewo;
    begin if pwb>5 then pwb:=pwb-1 else pwb:=16 end;

procedure w_gore;
    var nw1:byte;
    begin na_segmenty;
        if nw>1 then begin wysw_linie(nw,0); nw:=nw-1;nb:=nb-1 end
            else if nb>1 then
                begin wysw_linie(1,0);
                    nb:=nb-1;nw1:=tryb-ile_wierszy+1;
                    gotoxy(1,nw1-1);insline;
                    gotoxy(1,tryb+1);clreol;
                end;
                wysw_linie(nw,1)
    end;

procedure w_dol;
    var nw1:byte;
    begin
        if nb<999 then begin na_segmenty; nowe_wsp;
            if nw<ile_wierszy
                then begin wysw_linie(nw,0);nw:=nw+1;nb:=nb+1 end
                else begin wysw_linie(ile_wierszy,0);
                    nb:=nb+1;nw1:=tryb-ile_wierszy+1;
                    gotoxy(1,nw1);delline
                end;
            wysw_linie(nw,1);
            if nb>nb_max then nb_max:=nb;
            if nb>nb_max1 then nb_max1:=nb;
            if nb_max1>=poziom then
                begin zapis_kontrolny:=true;save;zapis_kontrolny:=false;
                    poziom:=poziom+poziom_zapisu
                end;
            end
    end;

procedure blad(nap:string);
    var poz:byte;
    begin
        poz:=100;
        okno_menu(poz,20,14,60,16,'',' @ '+nap+
            '@@',64+15,64+15);
    end;

procedure idz(blok:oddo);
    VAR I:oddo;
    begin
        if ((blok<=0) or (blok>999)) then
            begin blad(' ZLY NUMER BLOKU'); exit end;
        if blok>nb then for i:=nb to blok-1 do w_dol
            else for i:=nb downto blok+1 do w_gore
    end;

procedure wylancz_obroty;
    begin blad(' W Y L A N C Z O B R O T Y') end;

procedure pomiar;
    var m:string[16]; i:byte;
    begin
        if obraca_sie=false then
            begin

```

```

        m:='      0 00000 '+ob+' 8';
        for i:=5 to 16 do tabb[nb,i]:=m[i];
        w_dol
    end
else wylancz_obroty
end;

procedure korekta(ktora:typy_koriobr);
var m:string[16]; i:byte;
begin
    if obraca_sie=false then
        begin
            case ktora of
                wylanczamy :if gl=y then kb:='3' else kb:='5';
                wlanczamy:if gl=y then kb:='4' else kb:='6'
            end;
            m:='      '+kb+' 00000 99 1';
            for i:=5 to 16 do tabb[nb,i]:=m[i];
            w_dol
        end
    else wylancz_obroty
end;

procedure obroty(ktore:typy_koriobr);
var m:string[16]; i:byte;h:char;p,o:string[5];
const obr:boolean=true;
begin
    case ktore of
        wlanczamy : begin
            if obraca_sie=false then
                begin
                    zapamietaj_ekran;
                    zmaz(18,12,52,21,16+15,' ');
                    ramka(20,13,50,20);
                    kolor(podm,ekr1);
                    gotoxy(22,13);write('O B R O T Y');
                    kolor(podt,ekr1);
                    gotoxy(27,16);write('OBROTY (01-20) :');
                    ob:=lancuch(3);
                    if ob=#27 then begin odtworz_ekran; exit end;
                    if length(ob)=1 then ob:='0'+ob;
                    wb:='21000';obraca_sie:=true;
                    GOTOXY(27,17);write('POSOW (00-99) :');
                    pb:=lancuch(3);
                    if length(pb)=1 then pb:='0'+pb;
                    odtworz_ekran;
                    kolor(4,4);
                    podswietl(1,2,80,1)
                end
            else obr:=false
            end;
        wylanczamy :begin
            if obraca_sie=true then
                begin
                    pb:='00'; wb:='00000';obraca_sie:=false;
                    kolor(0,15);
                    podswietl(1,2,80,1)
                end
            else obr:=false
            end
        end;
    if obr=true then
        begin
            m:='      0 '+wb+' '+ob+' 7';
            for i:=5 to 16 do tabb[nb,i]:=m[i];
            w_dol
        end
    end
end;

```

```

        else wylancz_obroty;
        obr:=true
end;

procedure zmiana_bazy;forward;

procedure zmiana_wartosci(ktorej:wartosci);
var i,t1,t2:byte;
    znak:-1..1;
    m:string[16];wb1,napis:string[10];
    w1:real;
    ok:boolean;
begin
    if tabl[0,n]=0 then zmiana_bazy;
    t1:=24+ord(ktorej)*8;
    t2:=tryb-ile_wierszy+nw;
    zapamietaj_ekran;
    zmaz(t1,t2-1,t1+9,t2+1,15,' ');
    ramka(t1,t2-1,t1+9,t2+1);
    gotoxy(t1+3,t2);realread(w1);
    odtworz_ekran;
    if ((w1>999.99) or (w1<-999.99))
        then begin
            blad('  P O Z A  Z A K R E S E M');
            w_dol;w_gore;exit
        end
        else w1:=w1;
    case ktorej of
        x:begin if w1>tabl[nb-1,x] then kb:='1' else kb:='2';
            w1:=w1-tabl[nb-1,x];
            str((int((abs(w1)+0.000001)*100)+100000):14,wb1);
            wb:=copy(wb1,4,5)
        end;
        y:begin if w1>tabl[nb-1,y] then kb:='3' else kb:='4';
            w1:=w1-tabl[nb-1,y];
            str((int((abs(w1)+0.000001)*100)+100000):14,wb1);
            wb:=copy(wb1,4,5)
        end;
        z:begin if w1>tabl[nb-1,z] then kb:='5' else kb:='6';
            w1:=w1-tabl[nb-1,z];
            str((int((abs(w1)+0.000001)*100)+100000):14,wb1);
            wb:=copy(wb1,4,5)
        end;
        c:begin if w1>tabl[nb-1,c] then kb:='7' else kb:='8';
            w1:=w1-tabl[nb-1,c];
            str((int((abs(w1)+0.000001)*100)+100000):14,wb1);
            wb:=copy(wb1,5,5)
        end;
        n:tabl[nb,n]:=w1
    end;
    if ktorej<>n then m:=' '+kb+' '+wb+' 99 1'
        else m:=' 0 00000 '+ob+' 9';
    for i:=5 to 16 do tabb[nb,i]:=m[i];
    if nb>0 then w_dol
end;

procedure zmiana_narzedzia;
var m:string[16]; i:byte;
begin
    if obraca_sie=false then zmiana_wartosci(n)
        else wylancz_obroty
end;

procedure zmiana_bazy;
var i:integer;
    t,t1,t2:byte;
    znak:char;

```

```

j:wartosci;
const wart:string[5]='XYZCN';
begin while keypressed do znak:=readkey;
  repeat w_gore until nb=1;
  nw:=1;nb:=1;
  t:=tryb-ile_wierszy+nw;
  zapamietaj_ekran;
  zmaz(21,t-3,69,t+4,16+15,' ');
  ramka(23,t-2,67,t+3);
  kolor(podm,ekrl);
  gotoxy(25,t-2);write('Z M I A N A      B A Z Y');
  kolor(podt,ekrl);
  gotoxy(25,t+2);write('          GLOWICA PIONOWA-z POZIOMA-y ');
  gotoxy(60,t+2);znak:=readkey;
  case znak of
    'y','Y':gl:=y;
    'z','Z':gl:=z
  else begin odtworz_ekran;exit end;
end;
gotoxy(30,t+2);write('          GLOWICA '+znak);
znak:=readkey;
for j:=x to n do
  begin
    t1:=24+ord(j)*8;
    t2:=tryb-ile_wierszy+nw;
    repeat
      gotoxy(30,t+2);write(' '+wart[ord(j)+1]);
      gotoxy(t1+3,t2);
      read(wl);
      until not((wl>999.99) or (wl<-999.99));
      tabl[0,j]:=wl;
      gotoxy(t1+1,t2);write(wl:7:2)
    end;
    xbaz:=tabl[0,x];ybaz:=tabl[0,y];
    zbaz:=tabl[0,z];cbaz:=tabl[0,c];
    nbaz:=tabl[0,n];
    if gl=y then tabl[0,y]:=tabl[0,y]-nbaz
      else tabl[0,z]:=tabl[0,z]-nbaz;
    odtworz_ekran;
    nb:=1;nw:=1;w_dol;w_gore
  end;
end;

```

```

procedure load;
var numer,blok,blok1:string[16];
dap,blp:text;
ch:char;ok:boolean;
r,i,j:integer;k:real;
dan:array[0..40] of real;
begin
  for i:=0 to 40 do dan[i]:=0;
  zapamietaj_ekran;
  zmaz(18,13,57,19,16+15,' ');
  ramka(20,14,55,18);
  kolor(podm,ekrl);
  gotoxy(22,14);write('L O A D');
  kolor(podt,ekrl);
  gotoxy(25,16);write('PODAJ NAZWE PROGRAMU ');
  numer:=lancuch(7);
  odtworz_ekran;
  if numer='' then exit;
  IF (NUMER<>'0') and (numer<>#27) then
    begin
      assign(dap,sd+numer+rd);blok:=sd+numer+rd;
      if fileexist(blok)=true then
        begin
          reset(dap);
          readln(dap,blok);
        end;
    end;
end;

```



```

    if blok='BAZA' then
    begin
        read(dap,xbaz,ybaz,zbaz,cbaz,k);
        readln(dap);i:=0;readln(dap,blok);readln(dap,nbaz);
        if k=1 then
            begin gl:=y;tabl[0,y]:=ybaz-nbaz;tabl[0,z]:=zbaz end
        else begin gl:=z;tabl[0,z]:=zbaz-nbaz;tabl[0,y]:=ybaz end;
        repeat i:=i+1; readln(dap,dan[i]) until dan[i]=0;
        close(dap);
        tabl[0,x]:=xbaz;tabl[0,c]:=cbaz;tabl[0,n]:=nbaz
        end
    end
    else blad('    BRAK ZBIORU Z DANYMI');
    assign(blp,sb+numer+rb);
    if fileexist(sb+numer+rb)=true then
    begin
        reset(blp);
        repeat read(blp,ch) until ch='%';
        readln(blp);
        i:=0;r:=1;
        while not eof(blp) do
        begin
            i:=i+1; readln(blp,blok1);
            for j:=5 to 16 do tabb[i,j]:=blok1[j];
            if tabb[i,16]='9' then begin tabl[i,n]:=dan[r];r:=r+1 end
            end;
            close(blp); nb_max:=i
        end
    else blad('    BRAK ZBIORU');
    if (numer<>'0') and (numer<>#27) then
    begin
        nb:=1; nw:=1;
        for i:=1 to ile_wierszy do idz(i);
        for i:=ile_wierszy downto 1 do idz(i)
    end
end
end;

procedure druk;
var numer,symbol:string;
    dan:array[0..40] of real;
    wyniki:text;
    r,i,j:integer;
    ch,att,licznik:byte;
    h:char;

procedure prz;
begin licznik:=licznik+1;
    if licznik=63 then
    begin
        writeln(wyniki,#12);
        blad('    NASTEPNA STRONA - <RETURN>');
        licznik:=0
    end
end;

begin
    licznik:=0;
    zapamietaj_ekran;
    zmaz(18,13,57,20,16+15,' ');
    ramka(20,14,55,19);
    kolor(podm,ekrl);
    gotoxy(22,14);write('W Y D R U K');
    kolor(podt,ekrl);
    gotoxy(25,16);write(' NAZWA PROGRAMU ');
    numer:=lancuch(7);
    if numer=' ' then begin odtworz_ekran; exit end;

```

```

gotoxy(22,17);write(' SYMBOL CZESCI ');
symbol:=lancuch(30);
odtworz_ekran;
if (symbol<>'0') and (symbol<>#27) then
begin
  r:=2;
  for i:= 1 to nb_max do
    if tabb[i,16]='9' then
      begin
        dan[r]:=tabl[i,n];r:=r+1 end;
        dan[1]:=nbaz;
        assign(wyniki,'LPT1');rewrite(wyniki);
        writeln(wyniki,#27,#108,#8);
        writeln(wyniki,'
                                ',sb+numer+rb);prz;
        for i:=1 to 60 do write(wyniki,'*');writeln(wyniki);prz;
writeln(wyniki,'
                                L I S T A      N A R Z E D Z I');prz;
writeln(wyniki,'
                                - - - - -');prz;
writeln(wyniki,'
                                NR NARZ.          L NARZ');prz;
        for i:=1 to r-1 do
          begin
            writeln(wyniki,'
                                ',
            i,'
                                ',dan[i]:4:3);prz;
          end;
        for i:=1 to 60 do write(wyniki,'*');
        writeln(wyniki);prz;
        write(wyniki,'
                                BAZA          X');
        write(wyniki,'
                                Y');
        write(wyniki,'
                                Z');
        writeln(wyniki,'
                                C');prz;
        write(wyniki,'
                                ');
        write(wyniki,tabl[0,x]:4:3);write(wyniki,' ');
        write(wyniki,tabl[0,y]:4:3);write(wyniki,' ');
        write(wyniki,tabl[0,z]:4:3);write(wyniki,' ');
        writeln(wyniki,tabl[0,c]:4:3);prz;
        writeln(wyniki,'
                                NARZ. NR 1 L=',dan[1]:4:3);prz;
        writeln(wyniki,'DEC-',numer,'
                                SYMBOL-',symbol,'
                                ');prz;
        { writeln(wyniki,'
                                DATA ');
        }
        { writeln(wyniki,'%'); }
        nb:=1;nw:=1;r:=1;
        for i:=1 to ile_wierszy do wysw_linie(i,0);
        wysw_linie(1,1); w_dol;
        repeat
          for j:=1 to 55 do
            begin
              getchar(j,tryb-ile_wierszy+nw-1,ch,att);
              write(wyniki,chr(ch))
            end;
          writeln(wyniki);prz;
          w_dol;
          if tabb[nb-1,16]='9' then
            begin
              r:=r+1;
              writeln(wyniki,'
                                NARZ. NR ',
              r,'
                                L=',dan[r]:4:3);prz;
            end;
          until tabb[nb-2,16]='0';
          writeln(wyniki,#12);close(wyniki)
        end
      end;
end;

procedure save;
var numer,blok:string[16];
    ch:char;
    dap,blp:text;
    r,i,j:integer;k:real;
begin
  if zapis_kontrolny=false then

```

```

begin
  zapamietaj_ekran;
  zmaz(18,13,57,19,16+15,' ');
  ramka(20,14,55,18);
  kolor(podm,ekrl);
  gotoxy(22,14);write('S A V E');
  kolor(podt,ekrl);
  gotoxy(25,16);write('PODAJ NAZWE PROGRAMU ');
  numer:=lancuch(7);
  odtworz_ekran;
  if numer=' ' then exit
end
else numer:='000';
IF (numer<>'0') and (numer<>#27) then
  begin
    assign(dap,sd+numer+rd);
    if (fileexist(sd+numer+rd)=true) and (numer<>'000') then
      begin
        zapamietaj_ekran;
        zmaz(18,13,62,19,16+15,' ');
        ramka(20,14,60,18);
        kolor(podt,ekrl);
        gotoxy(25,16);write('ZBIOR ISTNIEJE ZASTAPIC (t/n)?');
        ch:=readkey;ch:=upcase(ch);
        odtworz_ekran;
        if ch<>'T' then exit
      end;
    rewrite(dap);
    if gl=y then k:=1 else k:=0;
    writeln(dap,'BAZA');
    write(dap,xbaz:8:2,ybaz:8:2,zbaz:8:2,cbaz:8:2,k:8:2);
    writeln(dap);writeln(dap,'NARZEDZIA');
    writeln(dap,nbaz:8:2);
    for i:=1 to nb_max do
      if tabb[i,16]='9' then writeln(dap,tabb[i,n]:8:2);
    i:=0;k:=0;
    writeln(dap,k:2:0);
    close(dap);
    assign(blp,sb+numer+rb);rewrite(blp);
    writeln(blp,'DEC-',numer);writeln(blp,'          %');
    repeat i:=i+1; writeln(blp,tabb[i]) until tabb[i,16]='0';
    close(blp)
  end
end;

procedure reset1;
var znak:char; i:integer; m:string[16]; j:byte;w:wartosci;
begin
  zapamietaj_ekran;
  zmaz(18,12,57,20,16+15,' ');
  ramka(20,13,55,19);
  kolor(podm,ekrl);
  gotoxy(22,13);write('R E S E T');
  kolor(podt,ekrl);
  gotoxy(25,16);write('WYKONAC RESET (t/n)');
  znak:=readkey;
  if (znak='t') or (znak='T') then
    begin
      gotoxy(22,17);write(' ZAPISAC BLOKI NA DYSK (t/n)');
      znak:=readkey;
      odtworz_ekran;
      if (znak='t') or (znak='T') then save;
      for i:=1 to nb_max do
        begin
          m:='    0 00000 00 0';
          for j:=5 to 16 do tabb[i,j]:=m[j]
        end;
      end;
    end;
end;

```

```

    nw:=1;nb:=1;nb_max:=2;nb_max1:=2;poziom:=50;
    for i:=1 to ile_wierszy do
        begin
            wysw_linie(i,0);
            for w:=x to t do tabl[i-1,w]:=0;
            end;
            for i:=1 to ile_wierszy do idz(i);
            for i:=ile_wierszy downto 1 do idz(i)
            end
        else odtworz_ekran;
    end;

procedure system1;
var znak:char;
begin
    zapamietaj_ekran;
    zmaz(18,12,57,20,16+15,' ');
    ramka(20,13,55,19);
    kolor(podm,ekrl);
    gotoxy(22,13);write('K O N I E C');
    kolor(podt,ekrl);
    gotoxy(25,16);write('OPOSCIC PROGRAM (t/n)');
    znak:=readkey;
    if (znak='t') or (znak='T') then
        begin
            gotoxy(22,17);write(' ZAPISAC BLOKI NA DYSK (t/n)');
            znak:=readkey;
            odtworz_ekran;
            if (znak='t') or (znak='T') then save;
            halt
        end
    else odtworz_ekran;
end;

procedure wstawianie_blokow;
var m:string[16];i,j,nb1,ile:integer;
begin
    zapamietaj_ekran;
    zmaz(18,11,63,20,16+15,' ');
    ramka(20,12,61,19);
    kolor(podm,ekrl);
    gotoxy(22,12);write('W S T A W I A N I E');
    kolor(podt,ekrl);
    gotoxy(27,15);write('POD KTORY BLOK (0-999) ');intread(nb1);
    gotoxy(27,16);write('ILE BLOKOW (1-255) ');intread(ile);
    odtworz_ekran;
    if ((ile<1) or (ile>255)) then
        begin blad(' ZA MALO BLOKOW'); exit end;
    if ((nb1<0) or (nb1>999)) then
        begin blad(' ZLY NUMER BLOKU'); exit end;
    for i:=nb_max downto (nb1+1) do
        begin
            for j:=5 to 16 do tabb[i+ile,j]:=tabb[i,j];
            tabl[i+ile]:=tabl[i]
        end;
    m:=' 0 00000 00 0';
    for i:=(nb1+1) to (nb1+ile) do
        for j:=5 to 16 do tabb[i,j]:=m[j];
    nb_max:=nb_max+ile;if nb1=0 then nb1:=1;
    if nb1<nb then idz(nb1);
    if nb1>=nb then
        begin idz(nb1+ile);idz(nb-ile) end
end;

procedure kasowanie_blokow;
var m:string[16];i,j,k,nb1,nb2,ile:integer;
begin

```

```

zapamietaj_ekran;
zmaz(18,11,63,20,16+15,' ');
ramka(20,12,61,19);
kolor(podm,ekrl);
gotoxy(22,12);write('M A Z A N I E');
kolor(podt,ekrl);
gotoxy(27,15);write('PIERWSZY BLOK (1-999) ');intread(nb1);
gotoxy(27,16);write('OSTATNI BLOK (1-999) ');intread(nb2);
odtworz_ekran;
ile:=nb2-nb1;k:=0;
if ((nb1<=0) or (nb2<=0) or (nb1>999) or (nb2>999)) then
  begin blad(' ZLY NUMER BLOKU'); exit end;
if (ile<0) then
  begin blad(' ZA MALO BLOKOW'); exit end;
for i:=(nb2+1) to nb_max do
  begin
    for j:=5 to 16 do tabb[nb1+k,j]:=tabb[i,j];
    tabl[nb1+k]:=tabl[i];k:=k+1
  end;
m:=' 0 00000 00 0';
for i:=nb_max downto (nb_max-ile+2) do
  for j:=5 to 16 do
    tabb[i,j]:=m[j];
nb_max:=nb_max-ile+2;
if nb1<nb then begin idz(nb1) end;
if nb1>=nb then
  begin ile:=nb;idz(nb+ile_wierszy-nw);idz(ile) end
end;

```

```

procedure kopiowanie_blokow;
var pie,ost,nb1,i,j,k:integer;m:string[12];
begin
  zapamietaj_ekran;
  zmaz(18,11,63,20,16+15,' ');
  ramka(20,12,61,19);
  kolor(podm,ekrl);
  gotoxy(22,12);write('K O P I O W A N I E');
  kolor(podt,ekrl);
  gotoxy(27,15);write('PIERWSZY BLOK (1-999) ');intread(pie);
  gotoxy(27,16);write('OSTATNI BLOK (1-999) ');intread(ost);
  gotoxy(27,17);write('POD KTORY BLOK (0-999) ');intread(nb1);
  odtworz_ekran;
  k:=1;
  if ((PIE<1) or (ost<1) or (nb1<0) or (pie>999) or (ost>999) or (nb1>999))
    then begin blad(' ZLY NUMER BLOKU'); exit end;
  if ((ost-pie)<0) then
    begin blad(' ZA MALO BLOKOW'); exit end;
  for i:=pie to ost do
    begin
      for j:=5 to 16 do tabb[nb1+k,j]:=tabb[i,j];k:=k+1;
      if tabb[nb1+k,16]='9' then tabl[nb1+k,n]:=tabl[i,n]
    end;
  if nb1=0 then nb1:=1;
  if nb1<nb then begin idz(nb1) end;
  if nb1>=nb then
    begin pie:=nb;idz(nb+ile_wierszy-nw);idz(pie) end
end;

```

```

procedure idz_do_bloku;
var blok,i:integer;
begin
  zapamietaj_ekran;
  zmaz(18,11,63,20,16+15,' ');
  ramka(20,12,61,19);
  kolor(podm,ekrl);
  gotoxy(22,12);write('I D Z D O B L O K U');
  kolor(podt,ekrl);

```

```

gotoxy(26,16);write('PODAJ NUMER BLOKU (1-999) ');
intread(blok);
odtworz_ekran;
idz(blok)
end;

procedure cykl;
var znak:char;
    sred,dziub,baza,dlugosc,skok,odskok,ile_razy,powrot,pozycja,
    dziub1,do_wb,poz_wrz:real;
    pb1:string[2];
    n1:byte;

function kier1:char;
begin if gl=y then kier1:='3' else kier1:='5' end;

function kier2:char;
begin if gl=y then kier2:='4' else kier2:='6' end;

procedure blokdod;
var ww:string[16];i:byte;
begin do_wb:=do_wb+0.000001;
    str(((do_wb*100)+100000):14,ww);wb:=copy(ww,4,5);
    ww:=' '+kb+' '+wb+' '+pb1+' 1';
    for i:=5 to 16 do tabb[nb,i]:=ww[i];ile:=ile+1;w_dol
end;
procedure wyciaganie_wior;
begin while poz_wrz<dlugosc-skok do
    begin
        kb:=kier1;do_wb:=poz_wrz;pb1:='99';blokdod;
        kb:=kier2;do_wb:=poz_wrz-0.5;pb1:='99';blokdod;
        kb:=kier2;do_wb:=skok+0.5;pb1:=pb;blokdod;
        poz_wrz:=poz_wrz+skok
    end;
    kb:=kier1;do_wb:=poz_wrz;pb1:='99';blokdod;
    kb:=kier2;do_wb:=poz_wrz-0.5;pb1:='99';blokdod;
    kb:=kier2;do_wb:=dlugosc-poz_wrz+dziub+0.5;pb1:=pb;blokdod;
    kb:=kier1;do_wb:=dlugosc+dziub+pozycja+powrot;pb1:='99';blokdod
end;

procedure lamanie_wior;
begin while (n1<ile_razy) and (poz_wrz<dlugosc-skok) do
    begin
        kb:=kier1;do_wb:=odskok;pb1:=pb;blokdod;
        kb:=kier2;do_wb:=skok+odskok;pb1:=pb;blokdod;
        poz_wrz:=poz_wrz+skok;n1:=n1+1
    end;
    if n1=ile_razy
    then wyciaganie_wior
    else begin kb:=kier1;do_wb:=odskok;pb1:=pb;blokdod;
        kb:=kier2;do_wb:=dlugosc-poz_wrz+odskok;pb1:=pb;blokdod;
        kb:=kier1;do_wb:=dlugosc+dziub+pozycja+powrot;pb1:='99';blokdod
        end
end;
procedure poczatek1;
begin if pozycja>10
    then begin kb:=kier2;do_wb:=pozycja-2;pb1:='99';blokdod;
        kb:=kier2;do_wb:=skok+2+dziub;pb1:=pb;blokdod;
        end
    else begin kb:=kier2;do_wb:=pozycja+skok+dziub;pb1:=pb;blokdod end
end;

begin
sred:=0;dziub:=0;
if razem_z_dziobem=true then
begin

```

```

zapamietaj_ekran;
zmaz(18,13,63,20,16+15,' ');
ramka(20,14,61,19);
kolor(podt,ekrl);
gotoxy(22,16);write('      STOZEK (t/n)');
znak:=readkey;
if (znak='t') or (znak='T') then
begin
gotoxy(22,16);write('  SREDNICA STOZKA = ');
realread(sred);
if sred<0 then begin blad('  UJEMNA WARTOSC ? !!!'); exit end;
dziub:=(pi/6)*(sred/2);dziub:=sin(dziub)/cos(dziub);
dziub:=abs(dziub)
end;
odtworz_ekran;
if znak=#27 then exit
end;
zapamietaj_ekran;
zmaz(33,13,63,25,16+15,' ');
ramka(35,14,61,24);
kolor(podm,ekrl);
gotoxy(37,14);write('C Y K L');
kolor(podt,ekrl);
gotoxy(41,16);write('BAZA = ');realread(baza);
if baza>tabl[nb-1,gl] then begin blad('  BAZA ZA WYSOKO');odtworz_ekran; exit end;;
gotoxy(41,17);write('DLUGOSC = ');realread(dlugosc);
if (dlugosc<=0) then
begin blad(' UJEMNA WARTOSC lub ZERO');odtworz_ekran; exit end;
if dlugosc<skok then begin blad('  ZA DUZY SKOK');odtworz_ekran; exit end;
gotoxy(41,18);write('SKOK = ');realread(skok);
if (skok<=0) then
begin blad(' UJEMNA WARTOSC lub ZERO');odtworz_ekran; exit end;
gotoxy(41,19);write('ODSKOK = ');realread(odskok);
if (odskok<0) then
begin blad(' UJEMNA WARTOSC lub ZERO');odtworz_ekran; exit end;
gotoxy(41,20);write('ILE = ');realread(ile_razy);
if (ile_razy<0) then
begin blad(' UJEMNA WARTOSC lub ZERO');odtworz_ekran; exit end;
gotoxy(41,21);write('POWROT = ');realread(powrot);

n1:=1;pozycja:=abs(tabl[nb-1,gl]-baza);kop:=nb;ile:=0;
if ile_razy=0 then ile_razy:=100;
if dlugosc=skok then ile_razy:=1;
if (ile_razy=1) and (dlugosc=skok) then
begin
poczatek1;
kb:=kier1;do_wb:=pozycja+skok+dziub+powrot;pb1='99';blokdol
end
else begin
dziub1:=dziub;dziub:=0;poczatek1;dziub:=dziub1;
poz_wrz:=skok;
if odskok=0 then wyciaganie_wior else lamanie_wior
end
end;

procedure kopiuj_cykl;
var i,j:byte;ww:string[16];
begin for i:=1 to ile do
begin for j:=5 to 16 do tabb[nb,j]:=tabb[kop+i-1,j];
w_dol
end
end;

procedure directory;
var
zbior:searchrec;

```

```

droga:string;
znak:char;
i:byte;
p_okno:pointer;
operacja:boolean;
rozszerzenie:string[3];
rodzaj:string[14];

begin
operacja:=savewindow(1,1,80,25,true,p_okno);
framewindow(71,2,80,24,$1f,$1f,'');
window(72,3,79,23);textattr:=$1f;clrscr;
droga:=sb+'*'+rb;
findfirst(droga,0,zbior);
gotoxy(3,2);
with zbior do write(copy(name,1,(length(name)-4)));
i:=2;znak:='D';
while (doserror=0) and (znak='D') do
begin
repeat
with zbior do name:='';
findnext(zbior);
i:=i+1;
with zbior do
begin
gotoxy(3,i);
write(copy(name,1,(length(name)-4)));
end;
until i=20;
i:=1;
while keypressed do znak:=readkey;
znak:=readkey;
if not(znak in[#27,'D','K']) then znak:='D';
clrscr;
end;
window(1,1,80,25);
restorewindow(1,1,80,25,true,p_okno);
end;

```

```

procedure konfig;
var zb:text;napis,raz:string[30];
begin
assign(zb,'\\DECKEL\\konfig.dec');
if fileexist('\\DECKEL\\konfig.dec')=false then exit;
reset(zb);
read(zb,napis);readln(zb,tryb);
read(zb,napis);readln(zb,ile_wierszy);
read(zb,napis);readln(zb,poziom_zapisu);
read(zb,napis);readln(zb,tkorekcji);
read(zb,napis);readln(zb,tpomiaru);
read(zb,napis);readln(zb,tobrotow);
read(zb,napis);readln(zb,tzmiany narz);
read(zb,napis);readln(zb,sb);
read(zb,napis);readln(zb,sd);
read(zb,napis);readln(zb,rb);
read(zb,napis);readln(zb,rd);
read(zb,napis);readln(zb,ekrt);
read(zb,napis);readln(zb,ekrl);
read(zb,napis);readln(zb,ekr);
read(zb,napis);readln(zb,podt);
read(zb,napis);readln(zb,podl);
read(zb,napis);readln(zb,podr);
read(zb,napis);readln(zb,pods);
read(zb,napis);readln(zb,podn);
read(zb,napis);readln(zb,podm);
read(zb,napis);readln(zb,raz);

```



```

if raz='TAK'
  then razem_z_dziobem:=true
  else razem_z_dziobem:=false;
close(zb)
end;

```

```

procedure podsw_w_pl(x,y:BYTE;VAR ch,att:byte);
begin
  kolor(podt,ekrl);
  podswietl(x+x1,y+y1,17,1);
  if x1=2 then x1:=22 else x1:=2;
  kolor(podm,podl);
  podswietl(x+x1,y+y1,17,1);
  getchar(x+x1,y+y1,ch,att)
end;

```

```

procedure podsw_w_gore(x,y:byte;var ch,att:byte);
begin
  kolor(podt,ekrl);
  podswietl(x+x1,y+y1,17,1);
  if y1>4 then y1:=y1-1 else y1:=16;
  kolor(podm,podl);
  podswietl(x+x1,y+y1,17,1);
  getchar(x+x1,y+y1,ch,att)
end;

```

```

procedure podsw_w_dol(x,y:byte;var ch,att:byte);
begin
  kolor(podt,ekrl);
  podswietl(x+x1,y+y1,17,1);
  if y1<16 then y1:=y1+1 else y1:=4;
  kolor(podm,podl);
  podswietl(x+x1,y+y1,17,1);
  getchar(x+x1,y+y1,ch,att)
end;

```

```

procedure meni_na_ekran(ktore:ktore_meni;x,y:byte);
var i:byte;

```

```

const n:array[1..34] of string[39]=
  ('***** PROCEDURY *****'),
  ('      ALT + LITERA  Z EDYTORA      '),
  ('      RETURN lub LITERA  Z KLAWIATURY  '),
  (' a -                n -  NARZEDZIE  '),
  (' b -  BAZA           o -  OBROTY WL  '),
  (' c -  ZMIANA C       p -  POMIAR     '),
  (' d -  DRUK           q -  KONIEC     '),
  (' e -  DEF CYKL       r -  RESET      '),
  (' f -  COPY CYKL      s -  SAVE       '),
  (' g -  COPY BLOKI     t -  OBROTY WYL  '),
  (' h -                u -             '),
  (' i -  IDZ DO         v -             '),
  (' j -  ODW. KOR.      w -  WSTAWIANIE  '),
  (' k -  KOREKCJA       x -  ZMIANA X    '),
  (' l -  LOAD           y -  ZMIANA Y    '),
  (' m -  MAZANIE        z -  ZMIANA Z    '),
  ('                    '),
  ('***** POMOC *****'),
  ('      SHIFT + LITERA  Z EDYTORA      '),
  ('      RETURN lub LITERA  Z KLAWIATURY  '),
  (' a -  KONFIG.        n -  NARZEDZIE  '),
  (' b -  BAZA           o -  OBROTY WL  '),
  (' c -  EDYTOR         p -  POMIAR     '),
  (' d -  DRUK           q -  KONIEC     '),
  (' e -  DEF CYKL       r -  RESET      '),
  (' f -  COPY CYKL      s -  SAVE       '),
  (' g -  COPY BLOKI     t -             '),

```

```

        (' h - KONWERSJA      u -          '),
        (' i - IDZ DO        v -          '),
        (' j - BLEDY         w - WSTAWIANIE '),
        (' k - KOREKCJA      x - X,Y,Z,C  '),
        (' l - LOAD          y - KLAWIATURA '),
        (' m - MAZANIE      z -          '),
        ('                    '));
begin
  window(x+1,y+1,x+40,y+17);
  kolor(podt,ekrl);
  case ktore of
    procedury:for i:=1 to 17 do begin gotoxy(1,i);write(n[i]) end;
    pomocy:for i:=18 to 34 do begin gotoxy(1,(i-17));write(n[i]) end
  end;
  window(1,1,80,25)
end;

procedure wywołanie_proc(znak:char); forward;
procedure wywołanie_pom(znak:char); forward;

procedure wysw_meni(ktore:ktore_meni;znak:char);
var kod,att,ch:byte; kursor:char;
const koniec2:boolean=false; x:byte=34;y:byte=4;
begin
  zapamietaj_ekran;
  zmaz(31,y-1,77,y+19,16+15,' ');
  ramka(33,y,75,y+18);
  meni_na_ekran(ktore,x,y);
  x1:=2;y1:=11;
  kolor(podm,podl);
  podswietl(x+x1,y+y1,17,1);
  if znak=#0 then
    repeat
      while keypressed do kursor:=readkey;
      kursor:=readkey;
      if kursor=#0 then
        begin
          kursor:=readkey;
          case kursor of
            #77,#75:podsw_w_pl(x,y,ch,att);
            #72:podsw_w_gore(x,y,ch,att);
            #80:podsw_w_dol(x,y,ch,att)
          end
        end
      else
        case kursor of
          #27:begin
            koniec2:=true;
            odtworz_ekran
          end;
          #13:begin
            koniec2:=true;
            odtworz_ekran;
            if ktore=pomocy
              then wywołanie_pom(char(ch))
              else wywołanie_proc(char(ch))
            end;
          'a'..'z':begin
            odtworz_ekran;
            if ktore=pomocy
              then wywołanie_pom(kursor)
              else wywołanie_proc(kursor);
            koniec2:=true
          end;
        end
      until koniec2=true;
  w_dol;w_gore;
end;

```

```

    koniec2:=false
end;

procedure wywołanie_proc(znak:char);
begin case znak of
    'a':direktory;
    'b':zmiana_bazy;
    'c':zmiana_wartosci(c);
    'd':druk;
    'e':cykl;
    'f':kopiuj_cykl;
    'g':kopiowanie_blokow;
    'h':begin end;
    'i':idz_do_bloku;
    'j':korekta(wylanczamy);
    'k':korekta(wlanczamy);
    'l':load;
    'm':kasowanie_blokow;
    'n':zmiana_narzedzia;
    'o':obroty(wlanczamy);
    'p':pomiar;
    'q':system1;
    'r':reset1;
    's':save;
    't':obroty(wylanczamy);
    'u':begin end;
    'v':begin end;
    'w':wstawianie_blokow;
    'x':zmiana_wartosci(x);
    'y':zmiana_wartosci(y);
    'z':zmiana_wartosci(z)
end
end;

procedure wywołanie_pom(znak:char);
var blok,zbior:string;
    plik:text;
    pos:integer;
    zn:char;i:byte;
begin
    pos:=0;
    zbior:='\DECKEL\HELP\' +znak+'.DEC';
    okno_zbior(pos,34,2,78,24,1,1,zbior,32+12,16+14);
end;

procedure podmeni (lista:typy_list);
var poz,x1,x2,y1,y2,j,k,l,m:byte;
    znacz,menu:string;
const koniec:boolean = false;
begin
    poz:=1;
    case lista of
        wewy:begin
            x1:=4;x2:=20;y1:=2;y2:=10;l:=7;
            znacz:='lsdarqb';
            menu := ' l    LOAD    @'+
                    ' s    SAVE    @'+
                    ' d    DRUK   @'+
                    ' a    DIR     @'+
                    ' r    RESET  @'+
                    ' q    KONIEC @'+
                    ' b    BAZA   @@';
        end;
        bloki:begin
            x1:=11;x2:=27;y1:=2;y2:=7;l:=4;
            znacz:='wmgi';
            menu:=' w    WSTAW   @'+

```

```

        ' m      KASUJ      @'+
        ' g      KOPIUJ     @'+
        ' i      IDZ DO     @@';
    end;
funkcje:begin
    x1:=19;x2:=37;y1:=2;y2:=9;l:=6;
    znacz:='pkjotn';
    menu:=' p      POMIAR      @'+
          ' k      KOREKTA     @'+
          ' j      ODW. KOR     @'+
          ' o      OBROTY      @'+
          ' t      WYL. OBR     @'+
          ' n      ZM. NARZ     @@';

    end
end;
okno_menu(poz,x1-1,y1,x2+1,y2-2,znacz,menu,64+14,16+15);
wywołanie_proc(znacz[poz]);
w_dol;w_gore;
end;

procedure poczatek;
const a:string[80]=' WE/WY      BLOKI      FUNKC.  X =      Y =      '+
                  ' Z =      CYKL      COPY      PROC.      POMOC ';
var i,j,k:integer;
    l:string[4]; w:wartosci;
begin
    kolor(ekrt,ekr);
    clrscr;konfig;
    for i:=1 to 80 do f[i]:=#32;
    for i:=0 to 999 do
        begin
            str((i+1000):4,1);
            tabb[i]:='000 0 00000 00 0';
            tabb[i,1]:=l[2];tabb[i,2]:=l[3];tabb[i,3]:=l[4];
            for w:=x to t do tabl[i,w]:=0
        end;
    kolor(podt,ekrl);
    for i:=0 to 9 do
        begin
            for j:= 1 to 8 do
                begin
                    k:=i*8+j;gotoxy(k,1);write(a[k])
                end;
            putchar(k-6,1,i+49,16+12)
        end;
    putchar(k-7,1,49,16+12);
    putchar(k-6,1,48,16+12);
    poziom:=poziom_zapisu;
    zapis_kontrolny:=false;
    nw:=1;nb:=1;nb_max:=1;nb_max1:=1;pwb:=5;pb:='11';ob:='11';
    ile:=1;
    obraca_sie:=false;
    for i:=1 to ile_wierszy do wysw_linie(i,0);
    wysw_linie(nw,1)
end;

const koniec:boolean=false;
var kursor:char;
begin poczatek;
    repeat
        while keypressed do kursor:=readkey;
        gotoxy(pwb,tryb-ile_wierszy+nw);
        kursor:=readkey;
        if kursor= #0
            then begin kursor:= readkey;
                    case kursor of

```

```

        #77:w_prawo;
        #75:w_lewo;
        #72:w_gore;
        #80:w_dol;
        #59:podmeni (wewy);
        #60:podmeni (bloki);
        #61:podmeni (funkcje);
        #62:zmiana_wartosci (x);
        #63:zmiana_wartosci (y);
        #64:zmiana_wartosci (z);
        #65:cykl;
        #66:kopiuuj_cykl;
        #67:wysw_meni (procedury,#0);
        #68:wysw_meni (pomocy,#0);
    #16..#50:wywołanie_proc (tablica [byte (kursor)])
end
end
else case kursor of
    #13:w_dol;
    #65..#90:wywołanie_pom (char (byte (kursor)+32));
    '0'..'9',#32:begin
        kolor (podr,pods); write (kursor);
        tabb[nb,pwb]:=kursor;w_prawo
        end;
        'x','X':zmiana_wartosci (x);
        'y','Y':zmiana_wartosci (y);
        'z','Z':zmiana_wartosci (z);
        'c','C':zmiana_wartosci (c)
    end;

until koniec=true;
end.

```

```

□
unit LW_OKNA;          {najnowsza wersja}
interface
uses dos,tpcrt;
const
    {adres_ekranu:longint=$b800;}
    {adres_linii :longint=$f00; }
    adres_ekranu=$b800;
    adres_linii =$f00;
type
    symbol=record
        znak:char;
        atrybut:byte;
    end;
    obraz =array[1..25,1..80] of symbol;
    wiersz=string[80];
var
    ekran:obraz absolute adres_ekranu:$0000;
    linia:array[1..80] of symbol absolute adres_ekranu:adres_linii;
    zapasowy:obraz;

    {***** GET- i PUTCHAR *****}
procedure getchar(x,y:byte;var chr,att:byte);

procedure putchar(x,y,chr,att:byte);

procedure okno_zbior(var poz:integer;
    x1,y1,x2,y2,w,k:byte;
    zbior:string;
    kolor1,kolor2:byte);
procedure okno_menu(var poz:byte;
    x1,y1,x2,y2:byte;
    znaczniki,menu:string;
    kolor1,kolor2:byte);

```

```

procedure podswietl(x1,y1,x2,y2,atr:byte);
procedure zmaz(x1,y1,x2,y2,atr:byte;zn:char);
procedure ramka(x1,y1,x2,y2:byte);
procedure ramka1(x1,y1,x2,y2,kolor1,kolor2:byte;napis:string);
procedure przesuwanie(var x1,y1,x2,y2:byte);
procedure zapamietaj_ekran;
procedure odtworz_ekran;
procedure cursor(status:boolean);
procedure obsluga_podswietlenia(var poz:byte;
                                x1,y1,x2,y2,kolor1,kolor2:byte);

```

```
implementation
```

```
{***** GET- i PUTCHAR *****}
```

```
procedure getchar(x,y:byte;var chr,att:byte);
```

```
begin
  chr:=byte(ekran[y,x].znak);
  att:=ekran[y,x].atrybut;
end;
```

```
procedure putchar(x,y,chr,att:byte);
```

```
begin
  ekran[y,x].znak :=char(chr);
  ekran[y,x].atrybut:=att;
end;
```

```
procedure zapamietaj_ekran;
```

```
begin
  move(ekran,zapasowy,4000);
end;
```

```
procedure odtworz_ekran;
```

```
begin
  move(zapasowy,ekran,4000);
end;
```

```
procedure cursor(status:boolean);
```

```
var reg:registers;
begin
  if status=true then normalcursor else hiddencursor;
end;
```

```
procedure podswietl(x1,y1,x2,y2,atr:byte);
```

```
var i,j,chr,att:byte;
begin
  for i:=x1 to x2 do
    for j:=y1 to y2 do
      begin
        getchar(i,j,chr,att);
        putchar(i,j,chr,atr);
      end;
end;
```

```
procedure obsluga_podswietlenia(var poz:byte;
```

```
                                x1,y1,x2,y2,kolor1,kolor2:byte);
```

```
var ile_wierszy:byte;
```

```
znak:char;
```

```
begin ile_wierszy:=y2-y1+1;
```

```
podswietl(x1,y1+poz-1,x2,y1+poz-1,kolor1);
```

```
repeat
```

```
  while keypressed do znak:=readkey;
```

```
  znak:=readkey;
```

```
  case znak of
```

```
    #0:begin
```

```
      znak:=readkey;
```

```
      podswietl(x1,y1+poz-1,x2,y1+poz-1,kolor2);
```

```

        case znak of
        {dol}      #72:if poz>1 then poz:=poz-1 else poz:=ile_wierszy;
        {gora}    #80:if poz<ile_wierszy then poz:=poz+1 else poz:=1;
        end;
        podswietl(x1,y1+poz-1,x2,y1+poz-1,kolor1)
        end;
        #27:poz:=0
        end;
until (znak=#27) or (znak=#13)
end;

```

```

function tlo(liczba:byte):byte;
begin case liczba of
        0:tlo:=0;
        1:tlo:=16;
        2:tlo:=32;
        3:tlo:=48;
        4:tlo:=64;
        5:tlo:=80;
        6:tlo:=96;
        7:tlo:=112
    end
end;

```

```

procedure zmaz(x1,y1,x2,y2,atr:byte;zn:char);
var i,j:byte;
begin
    for i:=x1 to x2 do
        for j:=y1 to y2 do
            begin
                putchar(i,j,byte(zn),atr);
            end
        end;
end;

```

```

procedure ramka(x1,y1,x2,y2:byte);
begin
    framewindow(x1,y1,x2,y2,$0f,$0f,'');
end;

```

```

procedure ramka1(x1,y1,x2,y2,kolor1,kolor2:byte;napis:string);
begin
    framewindow(x1,y1,x2,y2,kolor1,kolor2,napis);
end;

```

```

procedure przesuwanie(var x1,y1,x2,y2:byte);
var zapasowy:obraz;
    znak:char;
begin
    repeat while keypressed do znak:=readkey;
        move(ekran,zapasowy,4000);
        ramka(x1-1,y1-1,x2+1,y2+1);
        znak:=readkey;
        if znak=#0 then
            begin
                znak:=readkey;
                case znak of
                {prawo} #77:if x2<78 then begin x1:=x1+1;x2:=x2+1 end;
                {lewo}  #75:if x1>3  then begin x1:=x1-1;x2:=x2-1 end;
                {gora}  #72:if y1>2  then begin y1:=y1-1;y2:=y2-1 end;
                {dol}   #80:if y2<24 then begin y1:=y1+1;y2:=y2+1 end;
                {pgup}  #73:if y2-y1+1>1 then y2:=y2-1;
                {pgdn}  #81:if y2<24 then y2:=y2+1;
                {del}   #83:if x2<78 then x2:=x2+1;
                {ins}   #82:if x2-x1+1>2 then x2:=x2-1
                end
            end;
        move(zapasowy,ekran,4000);
end;

```

```

until (znak=#13) or (znak=#27);
end;

procedure okno_menu(var poz:byte;
                   x1,y1,x2,y2:byte;
                   znaczniki,menu:string;
                   kolor1,kolor2:byte);

var ile,chr,att:byte;
    znak:char;
    zapasowy:obraz;

procedure na_ekran;
var i,w,k:byte;
begin
    w:=0;i:=0;k:=1;
    while menu[k]<>'@' do
        begin
            while menu[k]<>'@' do
                begin
                    getchar(x1+i,y1+w,chr,att);
                    putchar(x1+i,y1+w,byte(menu[k]),att);
                    k:=k+1;i:=i+1
                end;
                k:=k+1;i:=0;w:=w+1
            end
        end;
    end;
begin
    cursor(false);
    move(ekran,zapasowy,4000);
    zmaz(x1-2,y1-1,x2+2,y2+1,kolor2,' ');
    ramka1(x1-1,y1-1,x2+1,y2+1,kolor2,kolor2,'');
    na_ekran;
    if poz>0 then podswietl(x1,y1+poz-1,x2,y1+poz-1,kolor1);
    repeat
        while keypressed do znak:=readkey;
        znak:=readkey;
        case znak of
            #0:if poz>0 then
                begin
                    znak:=readkey;
                    podswietl(x1,y1+poz-1,x2,y1+poz-1,kolor2);
                    case znak of
{dol}      #72:if poz>1 then poz:=poz-1 else poz:=y2-y1+1;
{gora}    #80:if poz<y2-y1+1 then poz:=poz+1 else poz:=1;
                    end;
                    podswietl(x1,y1+poz-1,x2,y1+poz-1,kolor1)
                end;
            #8:begin
                move(zapasowy,ekran,4000);
                przesuwanie(x1,y1,x2,y2);
                zmaz(x1-2,y1-1,x2+2,y2+1,kolor2,' ');
                ramka1(x1-1,y1-1,x2+1,y2+1,kolor2,kolor2,'');
                na_ekran;
                poz:=1;
                if poz>0 then podswietl(x1,y1+poz-1,x2,y1+poz-1,kolor1)
            end;
            #13:poz:=poz;
            #27:poz:=0
            else if poz>0 then
                begin
                    poz:=pos(znak,znaczniki);
                    if poz>0 then znak:=#13
                end
            end;
    until (znak=#27) or (znak=#13);
    move(zapasowy,ekran,4000);
    cursor(true)

```



```

end;

procedure okno_zbior(var poz:integer;
                    x1,y1,x2,y2,w,k:byte;
                    zbior:string;
                    kolor1,kolor2:byte);

const
  szer=80;
  dlugosc=600;
var tablica:array[1..dlugosc,1..szer] of char;
    plik:text;
    chr,att,wi,ko,ile_wierszy:byte;
    ile,dlug,li:integer;
    znak:char;
    zapasowy:obraz;
    przesun:boolean;

    procedure na_ekran;
    var i,j:byte;
        begin
            for j:=y1 to y2 do
                for i:=x1 to x2 do
                    begin
                        getchar(i,j,chr,att);
                        putchar(i,j,byte(tablica[ile+j-y1,k+i-x1]),att);
                    end;
                end;
            end;

    procedure lista;
    begin
        przesun:=true;
        podswietl(x1,y1+poz-1,x2,y1+poz-1,kolor2);
        case znak of
            #0:begin
                znak:=readkey;
                case znak of
                    #72:if poz>1 then begin przesun:=false;poz:=poz-1 end
                        else if ile>1 then ile:=ile-1;
                    #80:if poz<ile_wierszy then begin przesun:=false;poz:=poz+1 end
                        else if ile<dlug-ile_wierszy then ile:=ile+1;
                    #73:if ile-ile_wierszy>1
                        then ile:=ile-ile_wierszy
                        else ile:=1;
                    #81:if ile+ile_wierszy<=dlug-ile_wierszy
                        then ile:=ile+ile_wierszy
                        else ile:=dlug-ile_wierszy+1
                end
            end;
            #8:begin
                move(zapasowy,ekran,4000);
                przesuwanie(x1,y1,x2,y2);
                ile_wierszy:=y2-y1+1;
                zmaz(x1-2,y1-1,x2+2,y2+1,kolor2,' ');
                ramka1(x1-1,y1-1,x2+1,y2+1,kolor2,kolor2,'');
                na_ekran;
                poz:=1;
                podswietl(x1,y1+poz-1,x2,y1+poz-1,kolor1)
            end
        end;
        podswietl(x1,y1+poz-1,x2,y1+poz-1,kolor1);
    end;

    procedure tekst;
    begin
        case znak of
            #0:begin
                znak:=readkey;

```

```

        case znak of
            #72:if ile>1 then ile:=ile-1;
            #80:if ile<=dlug-ile_wierszy then ile:=ile+1;
            #73:if ile-ile_wierszy>1
                then ile:=ile-ile_wierszy
                else ile:=1;
            #81:if ile+ile_wierszy<=dlug-ile_wierszy
                then ile:=ile+ile_wierszy
                else ile:=dlug-ile_wierszy+1
        end
    end
end
end;

begin
    cursor(false);
    for wi:=1 to 80 do tablica[1,wi]:=' ';
    for ile:=1 to dlugosc do tablica[ile]:=tablica[1];
    assign(plik,zbior);
    {$I-} reset(plik); {$I+}
    if ioresult<>0
        then begin exit end
        else
            begin
                li:=1;
                repeat
                    ko:=1;
                    while not eoln(plik)do
                        begin
                            read(plik,tablica[li,ko]);ko:=ko+1
                        end;
                        readln(plik);li:=li+1;
                    until (li>dlugosc) or (eof(plik));
                    close(plik)
                end;
                dlug:=li-1;ile:=1;
                move(ekran,zapasowy,4000);
                zmaz(x1-2,y1-1,x2+2,y2+1,kolor2,' ');
                ramka1(x1-1,y1-1,x2+1,y2+1,kolor2,kolor2,'');
                ile_wierszy:=y2-y1+1;
                if poz<>0 then podswietl(x1,y1+poz-1,x2,y1+poz-1,kolor1);
                przesun:=true;
                repeat
                    if przesun=true then na_ekran;
                    while keypressed do znak:=readkey;
                    znak:=readkey;
                    if poz<>0 then lista else tekst
                until (znak=#27) or (znak=#13);
                if znak=#27 then poz:=0;
                if poz<>0 then poz:=ile+poz-1;
                move(zapasowy,ekran,4000);
                cursor(true)
            end;
        end;

    { if currentdisplay=monoherc then adres_ekranu:=$b000;}

    {*****}
end.

unit zab_LW;
interface
uses tpcrt;
implementation

procedure napisy;
var znak:char;
begin

```

```
writeln(' PROGRAM JEST ZABEZPIECZONY PRZED KOPIOWANIEM');
WRITELN(' JESLI CHCESZ GO URUCHOMIC NA INNYM KOMPUTERZE');
WRITELN(' ZGLOS SIE, CZEKAM !!! :');
WRITELN(' Waclaw Libront');
writeln(' 38-350 Bobowa 101');
writeln(' woj. Nowy Sacz');
```

```
end;
```

```
function czy_oli:boolean;
```

```
const
```

```
  a='04/';
```

```
  e1='kj3E3245 656jjÉ';
```

```
  b='LIVE';
```

```
  e='kjhklČyiuŁjjjoi';
```

```
  c='984';
```

```
  d='3/8';
```

```
  data_rom:string[8]=a+'0'+d+'6';
```

```
  typ_komp:string[13]='0'+b+'TTI 1'+c;
```

```
var
```

```
  i:word;
```

```
  czy:boolean;
```

```
begin
```

```
  for i:=1 to 8 do
```

```
    if chr(mem[$f000:$fff4+i])=data_rom[i]
```

```
      then czy_oli:=true
```

```
      else begin czy_oli:=false;exit end;
```

```
  for i:=1 to 13 do
```

```
    if chr(mem[$f000:$c04f+i])=typ_komp[i]
```

```
      then czy_oli:=true
```

```
      else begin czy_oli:=false;exit end;
```

```
end;
```

```
begin
```

```
  if (kbdflags and 3)=3 then exit;
```

```
  if paramstr(1)='W_L' then exit;
```

```
  if czy_oli then exit;
```

```
  napisy;
```

```
  halt(5);
```

```
end.
```

```
{*****}
```