

**LAMPIRAN-LAMPIRAN**



```

Program SOUND;
{$I-,R-}
{$M 17500,0,655360}

```

```

Uses Crt, Dos, Printer, graph;

```

```

Type PScr = Array [1..25] of pointer;
isi = Array [1..5] of String;
pos = Array [1..5] of byte;
kar = String [5];
vektor = array[1..4000] of byte;

```

```

Const
sti = $FB;

```

```

Var t : Char;
i, xtype : Byte;
data : vektor;
waktu, dt, mn, jm,
jim, min, dit, mdit, c : integer;
jani, meni, deti, d100, dd, jam, men, det, mdet : word;
din : boolean;
xmin, ymin, xmax, ymax: real;

```

```

{
-----
| Deklarasi Fungsi dan Prosedur |
-----
}

```

```

Function Pangkat(X, Y : Integer) : Integer;
Var i, z : Integer;
Begin
z := 1;
If y > 0 Then For i := 1 To Y Do z := z * X;
Pangkat := z;
End;

```

```

Function Des2Bin(Des : Integer) : String;
Var n, sisa : Integer;
s, Bin : String;
Begin
Bin := "";
s := "";
n := 0;
While Des >= 1 Do
Begin
If Des >= 2 Then sisa := Des Mod 2 Else sisa := Des;
Des := Des div 2;
inc(n);
Bin := Bin + chr (sisa + 48);
End;
for sisa := n Downto 1 Do s := s + Bin [sisa];
Des2Bin := copy ('00000000', 1, 8 - n) + s;
End;

```

```

Function Bin2Des(Bin:String):integer;
Var i,Des : integer;
Begin
Des := 0;
for i := 7 Downto 0 Do
If Bin [8 - i] = '1' Then Des := Des + pangkat (2, i);
Bin2Des := Des;
End;

```

```

Function Trim(s:String):String;
var i, j : Byte;
    st : String;
Begin
    i := Length (s);
    While s [i] = '' Do Dec (i);
    s [0] := Char (i);
    i := 1;
    While s [i] = '' Do Inc (i);
    st := "";
    For j := i To length(s) Do st := st + s [j];
    Trim := st
End;

Function Dup(X, ascii : Byte): String;
Var s : String;
Begin
    FillChar (s, SizeOf (s), chr (ascii));
    s [0] := Char (X);
    Dup := s
End;

Function Str2Num(s:String):real;
Var i, j, k : Integer;
    X : Real;
    st : String;
Begin
    i := Length (s);
    j := i;
    Repeat
        If s [j] = '' Then i := j - 1 Else j := j - 1;
        Dec (j)
    Until j = 0;
    if i = 0 then
        Begin
            Str2Num := 0;
            Exit
        End;
    k := 1;
    Repeat
        Inc(j);
        If s [j] = '' Then k := j + 1 Else j := j + 1;
    Until j = i;
    st := "";
    j := k - 1;
    Repeat
        Inc(j);
        If s[j] In ['0'..'9', '!'] Then st := st + s[j]
        Else
            Begin
                Str2Num := -1;
                Exit
            End
    Until j = i;
    Val('0' + st, X, k);
    Str2Num := X
End;

Function Num2Str(X : Real; w, D : Byte) : String;
Var s : String;
Begin
    Str(X, w, D, s);

```



```

Num2Str := s
End;

Function Scan (Y, X, p : Byte) : String;
Var reg : Registers;
    i : Byte;
    ss : String;
Begin
    ss [0] := Chr (p);
    For i := 1 To p Do
    Begin
        GotoXY(Y+i-1,X);
        reg.AH := 8;
        reg.BH := 0;
        Intr ($10, reg);
        ss [i] := Chr (reg.AL)
    End;
    Scan := ss
End;

procedure buka;
var    gd,gm,errorcode:integer;
begin
    gd:=detect;
    initgraph(gd,gm,"");
    errorcode:=graphresult;
    if errorcode <> Grok then halt;
end;

procedure Transformasi(x_min,y_min,x_max,y_max,x_gambar,y_gambar:real;
    var x_layar,y_layar:real);
var xmin,ymin,xmax,ymax:real;
begin
    xmin:=getmaxx div 10;
    xmax:=3*getmaxx div 4;
    ymin:=0;
    ymax:=8*getmaxy div 10;
    x_layar:=(xmin*x_max-xmax*x_min)/(x_max-x_min)
        +x_gambar*(xmax-xmin)/(x_max-x_min);
    y_layar:=(ymax*y_max-ymin*y_min)/(y_max-y_min)
        -y_gambar*(ymax-ymin)/(y_max-y_min);
end;

procedure skala(x_min,y_min,x_max,y_max:real);
var del,del_1,x1,y1,x2,y2:real;
    n:integer;
    dtx:string[5];
begin
    setcolor(white);
    Transformasi(x_min,y_min,x_max,y_max,x_min,y_min,x1,y1);
    Transformasi(x_min,y_min,x_max,y_max,x_min,y_max,x2,y2);
    line(trunc(x1),trunc(y1),trunc(x2),trunc(y2));
    setcolor(white);
    settextstyle(smallfont,horizdir,5);
    settextjustify(centertext,centertext);
    del:=(y_max-y_min)/5;
    del_1:=(y2-y1)/5;
    for n:=1 to 6 do
    begin
        if n= 6 then
            BEGIN

```

```

settextstyle(2,horizdir,6);
outtextxy(getmaxx div 20,trunc(y1+del_1*4.75),'(o));END
else
begin
settextstyle(smallfont,horizdir,5);
str(y_min+del*(n-1):5:2,dtx);
outtextxy(getmaxx div 20,trunc(y1+del_1*(n-1)),dtx);
if n<> 1 then
outtextxy(getmaxx div 10,trunc(y1+del_1*(n-1)),'-');
end;
end;
setcolor(white);
Transformasi(x_min,y_min,x_max,y_max,x_max,y_min,x2,y2);
line(trunc(x1),trunc(y1),trunc(x2),trunc(y2));
setcolor(white);

del:=(x_max-x_min)/5;
del_1:=(x2-x1)/5;
for n:=1 to 6 do
begin
if n= 6 then
BEGIN
settextstyle(smallfont,horizdir,9);
outtextxy(trunc(x1+del_1*4.75),17*getmaxy div 20-10,'ç');END
else
begin
settextstyle(smallfont,horizdir,5);
str(x_min+del*(n-1):5:2,dtx);
outtextxy(trunc(x1+del_1*(n-1)),17*getmaxy div 20,dtx);
if n<> 1 then
outtextxy(trunc(x1+del_1*(n-1)),8*getmaxy div 10-2,'l');
end;
end;
end;

procedure tutup;
begin
setcolor(white);
settextstyle(2,horizdir,6);
settextjustify(centertext,centertext);
outtextxy(getmaxx div 2,19*getmaxy div 20,'Tekan <Esc>');
repeat until readkey=#27;

end;

procedure kerangka;
begin
setcolor(white);
rectangle(0,0,3*getmaxx div 4,9*getmaxy div 10);
rectangle(3*getmaxx div 4,0,getmaxx,9*getmaxy div 10);
rectangle(0,9*getmaxy div 10,getmaxx,getmaxy);
setfillstyle(1,blue);
floodfill(1,1,white);
setfillstyle(1,lightblue);
floodfill(getmaxx-1,1,white);
setfillstyle(1,cyan);
floodfill(1,getmaxy-1,white);
setcolor(white);
settextstyle(2,horizdir,7);
settextjustify(centertext,centertext);
outtextxy(7*getmaxx div 8,getmaxy div 10,'Data');
end;

```

```

procedure tutup_layar;
var i,j:integer;
begin
  setcolor(black);
  for i:=0 to 1+getmaxx div 2 do
    begin
      line(i,0,i,getmaxy);
      line(getmaxx-i,0,getmaxx-i,getmaxy);
    end;
  end;

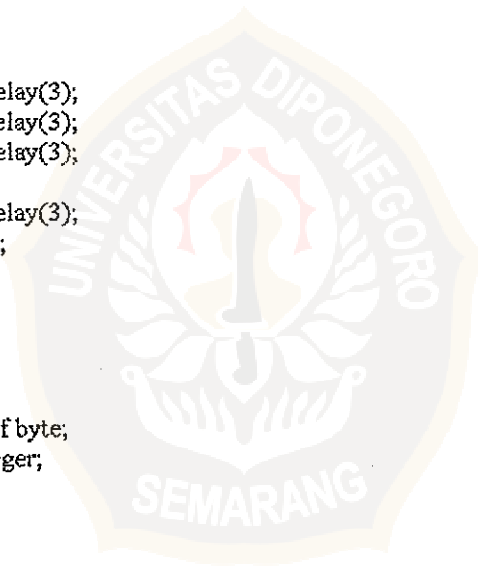
procedure inisialisasi;
begin
  clrscr;
  port[$300]:= $00;
  port[$301]:= $00;
  port[$302]:= $00;
  port[$303]:= $90;
end;

procedure ambilADC;
var
  data: byte;
  b: real;
begin
  port[$303]:= $90;
  port[$301]:= $05;delay(3);
  port[$302]:= $00;delay(3);
  port[$302]:= $01;delay(3);
  port[$302]:= $00;
  port[$302]:= $02;delay(3);
  DATA:=port[$300];
end;

procedure simpandata;
var
  w,waktu,tunda : file of byte;
  q,jumlah,sudut,y: integer;
  nama : string[15];
  b:real;

begin
  clrscr;
  Textcolor(14);
  writeln;
  write('banyaknya data yang diambil :');readln(jumlah);
  write('Nama Penyimpanan Data : ');
  readln(nama);
  assign(w,nama);
  rewrite(w);
  for q:=1 to jumlah do
    begin
      ambilADC;
      write(w,data[q]);
      {rekam data}
    end;
  writeln;
  close(w);
  write('data disimpan pada file : ',nama);

```



```

readln;
end;

procedure bacafile;
var
  data: vektor;
  w : file of byte;
  sudut,q,jumlah,tunda: integer;
  nama,waktu : string[15];
  b: real;
begin
  clrscr;writeln;
  write('file berisi data yang akan dibaca : ');
  readln(nama);
  assign(w,nama);
  reset(w);
  jumlah := filesize(w);
  for q:=1 to jumlah do
    begin
      read(w,data[q]);
      b:=((3.7556)*data[q])+(69.46);
      writeln(b);
    end;
  writeln;
  close(w);
  write('data dibaca dari file : ',nama);readln;
end;

procedure cetak;

var
  data: vektor;
  w : file of byte;
  q,jumlah: integer;
  nama : string[15];
  b:real;
Begin
  clrscr;
  write(' Bila printer siap tekan enter');
  readln;
  writeln;
  write('File berisi data yang akan dibaca : ');
  readln(nama);
  assign(w,nama);
  reset(w);
  jumlah := filesize(w);
  for q:=1 to jumlah do
    begin
      read(w,data[q]);
      b:=((3.7556)*data[q])+(69.46);
      writeln(1st,b);
    end;
  writeln(1st);
  close(w);
  write(1st,'data dibaca dari file : ',nama);
  write('Bila selesai tekan enter');
  readln;
end;

procedure tampilgrafik;

```

```

label awal;
var

```

```

del,del_1,x1,y1,x2,y2:real;
n:integer;
driver,mode,sudut,y,amp,A:integer;
x,sinus:real;
lagi,mau:char;
data:vektor;
w : file of byte;
q,jumlah,tunda: integer;
nama,waktu : string[15];
b:real;
begin
awal:
clrscr;
textcolor(yellow);
write('file berisi data yang akan dibaca : ');
readln(nama);

begin
buka;
xmin:=0;
xmax:=50;
ymax:=80;
ymin:=79;
kerangka;
skala(xmin,ymin,xmax,ymax);
assign(w,nama);
reset(w);
jumlah:=filesize(w);
for q:=1 to jumlah do
begin
read(w,data[q]);
b:=(3.7556)*data[q]+(69.46);
for sudut:=0 to jumlah do
begin
setcolor(15);
transformasi(xmin,ymin,xmax,ymax,sudut,trunc(b),x1,y1);
outtextxy(trunc(x1),trunc(y1),!);
delay(50);
end;
end;
tutup;
tutup_layar;
closegraph;
end;
close(w);
gotoxy(28,12);write('mau lagi (y/t) = ');readln(lagi);
if (lagi='y') or (lagi='t') then goto awal;
end;

procedure bacadata;
var tam: char;
begin
gotoxy(19,12);write('mau ditampilkan dalam bentuk grafik (y/t) : ');
readln(tam);
if (tam='y') or (tam='t') then
tampilgrafik
else
bacafile ;
end;

procedure ukur;
begin

```



```

clrscr;
BUKA;
xmin:=0;
xmax:=10;
ymax:=75;
ymin:=65;
kerangka;
skala(xmin,ymin,xmax,ymax);
transformasi(xmin,ymin,xmax,ymax, sudut,trunc(b), x1,y1);
outtextxy(trunc(x1),trunc(y1),!');
tutup;
tutup_layar;
closegraph;
textbackground(black);
textattr:=7;
end;

```

```

Procedure Cursor (i, j : Byte);
Var reg : Registers;
Begin
  reg.AH := 1;
  reg.CH := i;
  reg.CL := j;
  Intr ($10, reg)
End;

```

```

Procedure SaveScr (X1, X2 : Byte ; Var Line : PScr);
Var scr : pointer;
  n : Integer;
  i : Byte;
Begin
  n := 0;
  For i := X1 to X2 Do
  Begin
    GetMem (Line [i], 160);
    scr := Ptr ($B800, n);
    Move (scr^, Line [i]^, 160);
    Inc (n, 160);
  End
End;

```

```

Procedure LoadScr (X1, X2 : Byte ; Line : PScr);
Var scr : pointer;
  n : Integer;
  i : Byte;
Begin
  n := (X1 - 1) * 160;
  For i := X1 to X2 Do
  Begin
    scr := ptr ($B800, n);
    Move (Line [i]^, scr^, 160);
    Inc (n, 160);
    FreeMem (Line [i], 160)
  End
End;

```

```

Procedure ChAttr (X1, Y1, X2, Y2, Attr : Byte);
Var ij : Byte;
Begin

```

```

  For j := X1 to X2 do
  For i := Y1 to Y2 do
    Mem [$B800: 160 * j + 2 * i - 161] := Attr

```

```

End;

Procedure PutStr( Y, X, Attr : Byte ; s : String);
Var i : Byte;
Begin
  For i := 1 To Length(s) Do
    Begin
      Mem[$B800: (160 * (X - 1)) + (2 * (Y + i - 2))] := Ord(s[i]);
      Mem[$B800: (160 * (X - 1)) + (2 * (Y + i - 2)) + 1] := Attr
    End
  End;
End;

Function Input (X, Y, w, at1, at2 : Byte;
  hid : Boolean) : String;
Var i, j : ShortInt;
  st : String;
Begin
  j := Y;
  i := 0;
  ChAttr (X, Y, X, Y + w - 1, at1);
  TextAttr := at1;
  Cursor (6, 7);
  Repeat
    GotoXY (j + i, X);
    t := ReadKey;
    If (i = 0) And (t <> #13) Then PutStr (j, X, at1, Dup (w, 32));
    Inc (i);
    If t <> #0 Then
      Begin
        Case t Of
          #32..#255: If hid Then Begin Write (#254); st [i] := t End
          Else Write (t);
          #8 : Begin
            Dec (i, 2);
            If i < 1 Then i := 0;
            Putstr (j + i, X, at1, ' ');
          End;
          Else Dec (i)
        End
      End;
    If t = #0 Then
      Begin
        Case ReadKey Of
          #75 : If i = 1 Then i := 0 Else Dec (i, 2);
          #77 : If i > w Then i := w;
          Else Dec (i);
        End
      End;
  Until (i = w) Or (t In [#13, #27]);
  Cursor (21, 0);
  ChAttr (X, Y, X, Y + w, at2);
  st [0] := Char (i);
  If hid Then Input := st Else Input := Trim (Scan (Y, X, w));
End;

Procedure Box (X1, Y1, X2, Y2, at : Byte; db1, shw : Boolean);
Var a, b, c, D, e, f : Char;
  i : Byte;
Begin
  If shw Then ChAttr (X1 + 1, Y1 + 1, X2 + 1, Y2 + 1, 7);
  If db1 Then
    Begin
      a := #201; b := #205; c := #187; D := #186; e := #200; f := #188
    End

```

```

End
Else
Begin
  a := #218; b := #196; c := #191; D := #179; e := #192; f := #217
End;
PutStr (Y1, X1, at, a + Dup (Y2 - Y1 - 1, Ord (b) ) + c);
For i := 1 To X2 - X1 Do
  Putstr ( Y1, X1 + i, at, D + Dup (Y2 - Y1 - 1, 32) + D);
  PutStr (Y1, X2, at, e + Dup (Y2 - Y1 - 1, Ord (b) ) + f)
End;

Procedure Alarm;
VAR k : Integer;
Begin
  For k := 1000 To 5000 Do ;
  Nosound
End;

Procedure Alert (s : String);
Var p : Integer;
    scr : Pscr;
Begin
  SaveScr (1, 24, scr);
  p := 1 + (80 - Length (s) ) Div 2;
  Box (21, p - 2, 23, p + Length (s) + 1, 79, False, True);
  Putstr (p, 22, 79, s);
  Alarm;
  t := ReadKey;
  LoadScr (1, 24, scr);
  t := #0
End;

Procedure About;
Begin
  Box (7, 16, 16, 62, 95, False, True);
  Putstr (16, 7, 95, #213 + Dup (45, 205) + #184);
  Putstr (21, 8, 95, ' PRORAM PENGUKUR TINGKAT TEKANAN SUARA');
  Putstr (35, 9, 95, 'versi 1.00');
  Putstr (16, 10, 95, #195 + Dup (45, 196) + #180);
  Putstr (22, 11, 95, ' SLAMET HANDAYANTO AMIN');
  Putstr (33, 12, 95, 'J 401 94 1164');
  Putstr (16, 13, 95, #195 + Dup (45, 196) + #180);
  Putstr (19, 14, 95, ' JURUSAN FISIKA FMIPA ');
  Putstr (29, 15, 95, 'UNIVERSITAS DIPONEGORO');
  Putstr (16, 16, 95, #212 + Dup (45, 205) + #190);
  t := ReadKey;
End;

Procedure Shadow(X1, Y1, X2, Y2, Attr : Byte);
Var i : Byte;
Begin
  Putstr (Y2, X2 - 1, Attr, #220);
  For i := X2 To X1 Do PutStr (Y2, i, Attr, #219);
  For i := Y1 To Y2 Do PutStr (i, X1, Attr, #223)
End;

Procedure Button(s : String; Y, X, Attr1, Attr2 : Byte);
Begin
  Putstr (Y, X, Attr1, s);
  Shadow(X + 1, Y + 1, X + 1, Y + Length (s), Attr2)
End;

```

```

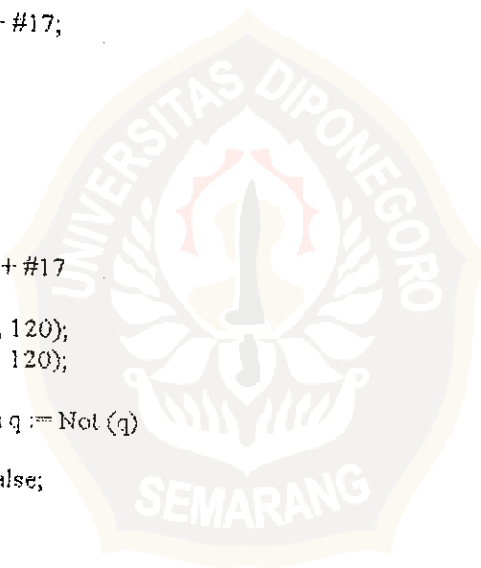
Procedure GoodBye;
Var i : Byte;
Begin
  TextAttr := 7;
  GotoXY (1, 1);
  For i := 1 To 25 Do DelLine;
  Cursor(6, 7);
  Halt (1)
End;

Procedure Quit;
Var Scrn : PScr;
  c1, c2 : Byte;
  s1, s2 : String;
  q : Boolean;
Begin
  SaveScr(1, 19, scrn);
  Box(11, 23, 18, 54, 112, True, True);
  Putstr(25, 13, 113, 'Ingin Keluar Dari Program ? ');
  q := true;
  Repeat
    if q Then
      Begin
        c1 := 79;
        c2 := 87;
        s1 := #16 + ' Ya ' + #17;
        s2 := ' Tidak '
      End
    ELSE
      Begin
        c1 := 87;
        c2 := 79;
        s1 := ' Ya ';
        s2 := #16 + ' Tidak ' + #17
      End;
    Button(s1, 28, 15, c1, 120);
    Button(s2, 41, 15, c2, 120);
    t := Readkey;
    If t In [#75, #77] Then q := Not (q)
    Until t In [#13, #27];
    If t = #27 Then q := False;
    Loadscr (1, 19, scrn);
    If q Then GoodBye;
    t := #7
  End;

Procedure Print;
Var Reg : Registers;
Begin
  reg.AH := 2;
  reg.DX := 0;
  Inr ($17, reg);
  If reg.AH And 8 = 8 Then
    Alert ('Terjadi kesalahan, coba periksa printernya...')
  Else
    End;

Procedure Tampilkan;
Var c, n : Integer;
  ok : Boolean;
Begin

```



```

If n = 0 Then
Begin
    Exit;
End;
c := 1;
Repeat

    PutStr (74, 15, 116, Nam2Str (c, 5, 0)),
    PutStr (64, 24, 116, #15 + ' F2 - Print ' + #17);
    t := ReadKey;
    If t = #0 Then t := ReadKey;
    Case t Of
        #71 : c := 1; {Horne}
        #79 : c := n; {End}
        #72, #73 : If c > 1 Then Dec (c); {Atas/PgUp}
        #80, #81 : If c < n Then Inc (c); {Bawah/PgDn}
        #60 : Print; {F2}
    End
Until t = #27;

End;

Function Menu (X1, Y1, X2, Y2, ATT, n : Byte; nmenu : isi;
              pmenu : pos; kmenu : kar) : Byte;
Var i, p : Byte;
Begin
    Box (X1, Y1, X2, Y2, ATT, False, True);
    For i := 1 To n Do
    Begin
        PutStr (Y1 + 2, X1 + i, ATT, nmenu [i]);
        PutStr (Y1 + pmenu[i] + 1, X1 + i, 30, kmenu [i]);
    End;
    p := 1;
    ChAttr (X1 + p, Y1 + 1, X1 + p, Y2 - 1, 116);
    Repeat
        t := ReadKey;
        ChAttr (X1 + p, Y1 + 1, X1 + p, Y2 - 1, ATT);
        PutStr (Y1 + pmenu[p] + 1, X1 + p, 30, kmenu [p]);
        For i := 1 To n Do If UpCase (t) = UpCase (kmenu [i]) Then
            Begin
                p := i;
                t := #13
            End;
        If t = #0 Then t := ReadKey;
        Case t Of
            #72 : If p > 1 Then Dec (p) Else p := n;
            #80 : If p < n Then Inc (p) Else p := 1;
            #75, #77 : p := 0;
        End;
        ChAttr (X1 + p, Y1 + 1, X1 + p, Y2 - 1, 116);
    Until t In [#13, #27, #75, #77];
    If t = #27 Then p := 0;
    Menu := p
End;

Procedure JudulMenu;
Begin
    PutStr (5, 2, 15, 'File ');
    ChAttr (2, 5, 2, 5, 14);
    ChAttr (2, 16, 2, 16, 14);

```

```
ChAttr (2, 27, 2, 27, 14);
End;
```

```
Procedure MenuUtama;
```

```
Const maxmenu = 1; {----Maksimum Jumlah Menu----}
```

```
lenmenu : Array [1..maxmenu] Of Byte = (5);
```

```
_menu : Array [1..maxmenu] Of isi =
```

```
(('Pengukuran', 'Simpan', 'Buka File', 'Cetak', 'Keluar'));
```

```
_pos : Array [1..maxmenu] Of pos
```

```
= ((1, 1, 1, 2, 1));
```

```
_kar : Array [1..maxmenu] Of Kar
```

```
= ('PSBeK');
```

```
Var p, p1, p2 : Byte;
```

```
pull : Boolean;
```

```
scrn1, scrn2 : PScr;
```

```
Begin
```

```
PutStr (1, 2, 7, Dup (80, 32));
```

```
JudulMenu;
```

```
SaveScr (1, 24, scrn1);
```

```
About;
```

```
LoadScr (1, 24, scrn1);
```

```
ChAttr (2, 5, 2, 8, 31);
```

```
pull := False;
```

```
p := 1;
```

```
Repeat
```

```
t := ReadKey;
```

```
If t = #27 Then Quit;
```

```
If t In [#13, 'F', 'D', 'f', 'd'] Then pull := true;
```

```
If t = #0 Then t := ReadKey;
```

```
Case t Of
```

```
#75 : If p > 1 Then Dec (p) Else p := maxmenu;
```

```
#77 : If p < maxmenu Then Inc (p) Else p := 1;
```

```
#80 : pull := True;
```

```
'F', 'f' : p := 1;
```

```
'D', 'd' : p := 2;
```

```
End;
```

```
JudulMenu;
```

```
ChAttr (2, 5 + 11 * (p - 1), 2, 4 + 11 * (p - 1) + lenmenu [p], 31);
```

```
If pull Then
```

```
Begin
```

```
Repeat
```

```
SaveScr (1, 25, scrn1);
```

```
p1 := Menu (3, 5 + 11 * (p - 1), 4 + length (_kar [p]),
```

```
8 + 11 * (p - 1) + Length (_menu [p, 1]), 23,
```

```
Length (_kar [p]), _menu [p], _pos [p], _kar [p]);
```

```
Case t Of
```

```
#75 : If p > 1 Then Dec (p) Else p := maxmenu;
```

```
#77 : If p < maxmenu Then Inc (p) Else p := 1;
```

```
End;
```

```
Case p Of
```

```
1 : Case p1 Of {MENU PILIHAN DSINI}
```

```
1 : UKUR ;
```

```
2 : SIMPANDATA;
```

```
3 : BACADATA;
```

```
4 : Cetak;
```

```
5 : Quit;
```

```
End;END;
```

```
LoadScr (1, 25, scrn1);
```

```
JudulMenu;
```

```
ChAttr (2, 5 + 11 * (p - 1), 2, 4 + 11 * (p - 1) + lenmenu [p], 31);
```

```
Until Not (t In [#75, #77]) And (p1 = 0);
```

```

    pull := False;
  End;
Until False;
End;

{-----}
|   Program Utama   |
{-----}

Begin
  ClrScr;
  Cursor (20, 0);
  {---Membuat Wallpaper-----}
  For i := 1 To 25 Do PutStr (1, 1, 31, Dup (80, 176));

  {---Menata Tampilan Menu Utama Program-----}
  PutStr (1, 1, 78, Dup (33, 32) + 'ALAT UKUR TINGKAT TEKANAN SUARA' + Dup
(33, 32) );
  PutStr (1, 25, 78, Dup (31, 32)+ '-----' + Dup (32, 32) );
  MenuUtama;
End.

```

