LAMPIRAN 1 : Program Fraktal Landscape

```
Program Landscape;

Uses CRT,Graph,Fractal;

var
        Triangle: array[1..3] of PointType;
          GraphDriver,GraphMode,a,i,j,k,l,m,level,row,col,key,ke
y1,y_max: integer;
          x,y,xz,yz: real;

Function Menu(kode: integer): integer;
Const
     Text1: array[0..5] of string [45] =
            (('Pengertian Fraktal Landscape'),
            ('Fraktal Landscape OAK CREEK CANYON'),
            ('Fraktal Landscape PIKE PEAK AT SUNRISE'),
            ('Fraktal Landscape EARTH VIEWED FROM THE MOON'),
            ('Programmer Fraktal Landscape'),
            ('Keluar'));
     Text2: array[0..5] of string [45] =
            (('Level 0'),('Level 1'),('Level 2'),('Level 3'),
            ('Level 4'),('Keluar ke Menu Utama'));
     Color: array[0..15] of integer =
(0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15);

Var
     i,k1: integer;
     ch1: char;
begin
     clrscr;
     k1:=0;
     Window(1,1,80,25);
     TextBackGround(3);

     if kode = 0 then
     begin
          Window(19,6,68,20);
          TextBackGround(4);
          Window(20,6,69,20);
          TextBackGround(3);
          Window(18,6,67,20);
          TextBackGround(0);
          l:=5;
          TextBackGround(0);
     end
     else
     begin
          Window(18,10,67,22);
          l:=5;
          TextBackGround(14);
          clrscr;
     end;

     {clrscr;}
     GotoXY(3,2);
     if kode = 0 then
     begin
          TextColor(14);
          Writeln('                SELAMAT DATANG   ');
          Writeln('            PROGRAM FRAKTAL LANDSCAPE');
          Write('
================================================');
     end
     else
     begin
          TextColor(15);
          Writeln;
          Writeln('
=============================================');
          Writeln;
     end;

     repeat
           for i:=0 to 5 do
           begin
                if i = k1 then
                begin
                     if kode=0 then
                     begin
                          TextColor(14);
                          TextBackGround(4);
                     end
                     else
                     begin
                          TextColor(1);
                          TextBackGround(10);
                     end
                end
                else
                begin
                     if kode=0 then
                     begin
                          TextColor(15);
                          TextBackGround(0);
                     end
                     else
                     begin
                          TextColor(1);
                          TextBackGround(14);
                     end;
                end;
                GotoXY(4,i+5);
                if kode = 0 then
                   Write(Text1[i])
                else
                   Write(Text2[i]);
           end;
           ch1:=ReadKey;
           if ch1 = char(#00) then
           begin
                ch1:=ReadKey;
                case ch1 of
                     'P':
                     begin
                        if kode = 0 then
                        begin
                             if k1=5 then
                             k1:=0
                             else
                             inc(k1);
                        end
                        else
                        begin
                             if k1=5 then
                             k1:=0
                             else
                             inc(k1);
                        end;
                     end;
                     'H':
                     begin
                        if kode=0 then
                        begin
                             if k1=0 then
                             k1:=5
                             else
                             dec(k1);
                        end
                        else
                        begin
                             if k1=0 then
                             k1:=5
                             else
                             dec(k1);
                        end;
                     end;
                end;
           end;
           until ch1=char($0D);
           Window(1,1,80,25);
           menu:=k1;
     end;

Procedure Info;
Const
     TextLand: array[0..8] of string [80] =
     ((' '),
     ('  '),
     ('                PENGERTIAN FRAKTAL LANDSCAPE'),

('============================================================'),
     ('  '),
     ('Fraktal Landscape adalah jenis visual fraktal yang '),
     ('memperlihatkan suatu permukaan ataupun pemandangan dari
'),
     ('sebuah obyek alami, misalnya pegunungan, bukit ataupun
'),
     ('permukaan bumi lainnya'));
     TextOak: array[0..8] of string [80] =
     (('  '),
     ('  '),
     ('             FRAKTAL LANDSCAPE OAK CREEK CANYON'),

('============================================================'),
     ('  '),
     ('Fraktal landscape ini memvisualisasikan suatu pemandangan
'),
     ('sebuah lembah atau ngarai di dekat Sedona, Arizona,
Amerika'),
     ('Serikat, yang sering disebut sebagai Oak Creek Canyon
atau'),
     ('Lembah Oak Creek'));
     TextPike: array[0..8] of string [80] =
     (('  '),
     ('  '),
     ('          FRAKTAL LANDSCAPE PIKES PEAK AT SUNRISE'),

('============================================================'),
     ('  '),
     ('Fraktal landscape pemandangan saat matahari terbit dari
'),
     ('beberapa puncak gunung di dekat Colorado, Amerika
Serikat,'),
     ('dan sering disebut dengan puncak Pike atau Pikes Peak'),
     ('  '));
     TextEarth: array[0..8] of string [80] =
     ((' '),
     (' '),
     ('     FRAKTAL LANDSCAPE EARTH VIEWED FROM THE MOON'),

('=========================================================='),
     (' '),
     ('Sebuah visualisasi yang indah dari pemandangan bumi
yang'),
     ('dilihat dari bulan, dengan berjuta bintang dan sebuah
benda'),
     ('angkasa yang menghiasinya'),
     (' '));
     TextFBI: array[0..8] of string [80] =
     ((' '),
     (' '),
     ('           PROGRAMMER FRAKTAL LANDSCAPE 2003'),

('=========================================================='),
     (' '),
     ('   NAMA PEMBUAT PROGRAM   : Febrian Widiarto'),
     ('   NOMOR INDUK MAHASISWA  : J2A 096 023'),
     ('   TEMPAT & TANGGAL LAHIR : Semarang, 9 Februari 1978'),
     (' '));
var
   ch2: char;
begin
{     if key>5 then
     begin
          TextBackGround(0);
          clrscr;
          Window(14,4,69,14);
          TextBackGround(1);
          clrscr;
          TextColor(14);j:=8;
     end
     else}
     begin
          TextBackGround(0);
          clrscr;
          Window(10,4,75,15);
          TextBackGround(1);
          clrscr;
          TextColor(14);j:=9;
     end;
     for i:=0 to j do
     begin
          GotoXY(3,i+2);
          if key=0 then write(TextLand[i]);
          if key=1 then write(TextOak[i]);
          if key=2 then write(TextPike[i]);
          if key=3 then write(TextEarth[i]);
          if key=4 then write(TextFBI[i]);
     end;
     ch2:=ReadKey;
     if (key=1) or (key=2) or (key=3) then
        key1:=menu(1);
end;

procedure generate(x1: integer; y1: integer; x2: integer; y2:
integer; x3: integer;
        y3: integer; level: integer; color1: integer; color2:
integer);forward;

procedure node(x1: integer; y1: integer; x2: integer; y2:
integer; x3: integer;
        y3: integer; x4: integer; y4: integer; x5: integer; y5:
integer; x6: integer;
        y6: integer; level: integer; color1: integer; color2:
integer);
begin
        if level<>0 then
        begin
        generate(x1,y1,x4,y4,x6,y6,level-1,color1,color2);
        generate(x6,y6,x5,y5,x3,y3,level-1,color1,color2);
        generate(x4,y4,x2,y2,x5,y5,level-1,color1,color2);
        generate(x4,y4,x4,y4,x6,y6,level-1,color1,color2);
        end;
end;

procedure midpoint(x: real; y: real);
var
          r,w: real;
          seed: longint;

begin
          seed := Round(350*y + x);
          RandSeed := seed;
          r := 0.33333 + Random/3.0;
          w := 0.015 + Random/50.0;
        if Random < 0.5 then
                    w := -w;
          xz := r*x - (w + 0.05)*y;
          yz := r*y + (w + 0.05)*x;
end;

procedure plot_triangle(x1: integer; y1: integer; x2: integer;
y2: integer;
          x3: integer; y3: integer; color1: integer; color2:
integer);
var
        ytt,color,a: integer;
        zt: real;
begin
        if y1 > y2 then
           ytt := y1
        else
           ytt := y2;
        if ytt < y3 then
           ytt := y3;
           zt := ABS(ytt/y_max);
        if Random <= zt then
                    color := color1
          else
                    color := color2;
        Triangle[1].x := x1 + 320;
          Triangle[1].y := 175 - (longint(93*y1) div 128);
          Triangle[2].x := x2 + 320;
          Triangle[2].y := 175 - (longint(93*y2) div 128);
          Triangle[3].x := x3 + 320;
          Triangle[3].y := 175 - (longint(93*y3) div 128);
          SetColor(color);
          SetFillStyle(1,color);
          FillPoly(3,Triangle);
end;

procedure generate(x1: integer; y1: integer; x2: integer; y2:
integer; x3: integer;
        y3: integer; level: integer; color1: integer; color2:
integer);
var
        x4,x5,x6,y4,y5,y6: integer;
begin
        x := x2 - x1;
        y := y2 - y1;
        midpoint(x,y);
        x4 := x1 + Round(xz);
        y4 := y1 + Round(yz);
        x := x3 - x1;
        y := y3 - y1;
        midpoint(x,y);
        x6 := x1 + Round(xz);
        y6 := y1 + Round(yz);
        x := x3 - x2;
        y := y3 - y2;
        midpoint(x,y);
        x5 := x2 + Round(xz);
        y5 := y2 + Round(yz);
        if level=0 then
           begin
                plot_triangle(x1,y1,x4,y4,x6,y6,color1,color2);
                plot_triangle(x2,y2,x4,y4,x5,y5,color1,color2);
                plot_triangle(x3,y3,x5,y5,x6,y6,color1,color2);
                plot_triangle(x4,y4,x5,y5,x6,y6,color1,color2);
           end
           else
           begin
                plot_triangle(x1,y1,x4,y4,x6,y6,color1,color2);
                plot_triangle(x2,y2,x4,y4,x5,y5,color1,color2);
                plot_triangle(x3,y3,x5,y5,x6,y6,color1,color2);
                plot_triangle(x4,y4,x5,y5,x6,y6,color1,color2);

node(x1,y1,x2,y2,x3,y3,x4,y4,x5,y5,x6,y6,level,color1,color2);de
lay(1);
           end
end;

procedure gen_quad(x1: integer; y1: integer; x2: integer; y2:
integer;
         x3: integer; y3: integer; x4: integer; y4: integer;
level: integer;
         color1: integer; color2: integer);
begin
       generate(x1,y1,x2,y2,x3,y3,level,color1,color2);
       generate(x1,y1,x4,y4,x3,y3,level,color1,color2);
end;

procedure cactus(x1: integer; y1: integer; scale: integer;
level: integer;
         color1: integer; color2: integer);
begin

gen_quad(x1,y1,x1,y1+21*scale,x1+2*scale,y1+22*scale,x1+2*scale,
            y1,level,color1,color2);

gen_quad(x1+2*scale,y1,x1+2*scale,y1+22*scale,x1+3*scale,
            y1+21*scale,x1+3*scale,y1,level,color1,color2);

gen_quad(x1,y1+9*scale,x1+7*scale,y1+9*scale,x1+7*scale,y1+12*sc
ale,
            x1,y1+12*scale,0,color1,color2);

gen_quad(x1,y1+9*scale,x1+6*scale,y1+9*scale,x1+7*scale,y1+12*sc
ale,
            x1,y1+12*scale,level,color1,color2);
```

```
gen_quad(x1+7*scale,y1+9*scale,x1+7*scale,y1+16*scale,x1+9*scale
,

y1+17*scale,x1+9*scale,y1+9*scale,level,color1,color2);

gen_quad(x1+9*scale,y1+9*scale,x1+9*scale,y1+16*scale,x1+10*scal
e,

y1+17*scale,x1+10*scale,y1+10*scale,level,color1,color2);
        gen_quad(x1,y1+7*scale,x1-6*scale,y1+7*scale,x1-
6*scale,y1+10*scale,
            x1,y1+10*scale,0,color1,color2);
        gen_quad(x1,y1+7*scale,x1-6*scale,y1+7*scale,x1-
6*scale,y1+10*scale,
            x1,y1+10*scale,level,color1,color2);
        gen_quad(x1-7*scale,y1+8*scale,x1-
7*scale,y1+12*scale,x1-6*scale,
            y1+13*scale,x1-
6*scale,y1+7*scale,level,color1,color2);
        gen_quad(x1-6*scale,y1+7*scale,x1-
6*scale,y1+13*scale,x1-4*scale,
            y1+12*scale,x1-
4*scale,y1+7*scale,level,color1,color2);
end;

Procedure Oak;
const
        color_set: array[0..15] of integer
=(27,1,6,10,4,5,2,7,56,57,58,59,4,61,
                    62,63);
          x1: array[0..25] of integer = (-330,-90,-
90,120,120,120,-160,-120,-120,
                 -80,-80,-50,-50,-
50,80,104,104,128,128,152,152,200,-470,-350,
                 -220,-200);
          y1: array[0..25] of integer = (-110,-110,-110,-110,-
110,-110,-10,-10,
                 -10,-10,-10,-10,-10,-
10,50,50,50,50,50,50,50,50,-300,-280,-280,
                 -280);
          x2: array[0..25] of integer = (-160,-160,0,0,80,200,-
160,-160,-120,-120,
                 -80,-80,-50,0,100,100,104,104,128,128,152,152,-
250,-60,80,230);
          y2: array[0..25] of integer =
(0,0,0,0,50,50,220,220,190,190,230,230,100,
                 180,180,180,200,205,215,215,160,160,-110,-140,-
130,-120);
          x3: array[0..25] of integer = (-90,0,120,80,200,340,-
120,-120,-80,-80,
                 -50,-
50,0,0,104,104,128,128,152,152,200,200,300,300,340,580);
          y3: array[0..25] of integer = (-110,0,-110,50,50,-
110,-10,220,-10,200,
                 -10,235,180,-10,50,200,50,210,50,220,50,140,-
300,-300,-300,-300);

begin
          GraphDriver := 4;
          GraphMode := EGAHi;
          InitGraph(graphDriver,GraphMode,'');
          SetColor(15);
        for i:=0 to 15 do
                    setEGApalette(i,color_set[i]);
        y_max := 480;
        for i:=0 to 21 do

generate(x1[i],y1[i],x2[i],y2[i],x3[i],y3[i],level,
                2,12);
        gen_quad(-330,-260,-330,-100,330,-100,330,-
260,level,14,14);
        cactus(-200,-120,2,level,2,6);
        cactus(-110,-130,3,level,2,6);
        cactus(0,-160,4,level,2,6);
        cactus(150,-200,6,level,2,6);
        readkey;readkey;CloseGraph;
end;

Procedure Pike;
const
        color_set: array[0..15] of integer
=(62,1,2,3,4,5,20,7,56,57,58,59,60,61,
                    62,63);
begin
          GraphDriver := 4;
          GraphMode := EGAHi;
          InitGraph(graphDriver,GraphMode,'');
          SetColor(15);
          for i:=0 to 15 do
                    setEGApalette(i,color_set[i]);
        y_max := 480;
          generate(-220,-240,120,100,500,-40,level,15,11);
          generate(-780,-200,0,30,420,-200,level,15,9);
          generate(-100,-250,240,40,500,-180,level,14,6);
          generate(-770,-175,-250,-75,600,-325,level,1,15);
          generate(-550,-270,-60,-140,400,-300,level,7,8);
          generate(-220,-280,80,-125,340,-300,level,12,15);
```

```
          generate(-200,-280,230,-150,580,-350,level,0,7);
        readkey;readkey;CloseGraph;
end;

Procedure Earth;
const
        color_set: array[0..15] of integer
=(0,1,2,3,4,5,20,7,56,57,58,59,60,61,
                    62,63);
          xa: array[0..32] of integer = (-82,-80,-90,-70,-50,-
30,-25,25,40,42,
                    20,35,40,50,60,60,-28,70, -
25,70,108,81,60,45,48,96,45,38,-8,0,
                    -20,-28,55);
          xb: array[0..32] of integer = (-70,-70,-80,-50,-
30,25,-25,40,65,65,
                    40,38,40,60,60,70,-28,90, -
20,105,92,70,56,48,54,100,38,46,12,
                    14,14,8,62);
          xc: array[0..32] of integer = (-70,-70,-80,-50,-
50,20,30,40,40,58,
                    40,37,50,50,70,75,20,90, -
40,95,83,70,45,60,60,54,106,65,12,
                    40,14,-8,95);
          xd: array[0..32] of integer = (-90,-80,-90,-70,-
30,20,20,25,50,50,
                    20,40,50,60,70,75,20,70, -
40,90,81,108,45,56,96,45,100,106,8,
                    44,0,-30,55);
          ya: array[0..32] of integer =
(52,52,76,80,76,38,10,80,90,55,50,3,
                    60,60,52,55,80,115, 38,109,76,80,-130,-124,-
90,-70,-60,-50,0,
                    -10,10,90,-100);
          yb: array[0..32] of integer =
(52,52,76,80,80,30,30,80,70,70,40,5,
                    8,52,38,38,20,120, 78,104,58,95,-124,-90,-
65,-60,-50,-25,0,
                    -10,10,80,-100);
          yc: array[0..32] of integer =
(60,80,80,55,56,38,10,90,80,70,3,-5,
                    60,20,38,40,20,106, 78,76,60,110,-124,-100,-
100,-65,-50,-25,
                    -18,-30,-18,75,-50);
          yd: array[0..32] of integer=
(60,80,77,55,38,30,30,90,60,60,3,-4,
                    20,27,38,40,74,109, 43,76,80,76,-124,-124,-
70,-60,-60,-50,
                    -18,-30,-18,85,-50);
        j: array[0..18] of integer =
(75,200,300,350,400,425,500,100,175,
                325,375,460,30,260,500,425,150,250,90);
        k: array[0..18] of integer =
(260,290,270,290,260,270,275,270,300,
                290,275,260,290,270,295,290,280,300,295);
        l: array[0..18] of integer =
(5,5,5,5,5,5,5,12,12,12,12,12,18,22,
                22,30,22,16,30);
        m: array[0..18] of integer =
(2,2,2,2,2,2,2,7,7,7,7,7,7,10,10,10,
                9,5,7);

var
        x_center,y_center: integer;

begin
          GraphDriver := 4;
          GraphMode := EGAHi;
          InitGraph(graphDriver,GraphMode,'');
          SetColor(15);
          for i:=0 to 15 do
                    setEGApalette(i,color_set[i]);
          x_center := -140;
          y_center := 30;
          for i:=0 to 2000 do
          begin
                    row := Random(350);
                    col := Random(640);
                    PutPixel(col,row,15);
          end;
          SetFillStyle(1,11);
          SetColor(11);
          FillEllipse(210,155,152,106);
          y_max := 480;
          for i:=0 to 32 do

          gen_quad(xa[i]+x_center,ya[i]+y_center,xb[i]+x_center,

          yb[i]+y_center,xc[i]+x_center,yc[i]+y_center,xd[i]+

          x_center,yd[i]+y_center,level,15,15);
        generate(-410,-300,-250,-100,300,-300,level,7,15);
          generate(-350,-280,-60,-110,300,-300,level,7,15);
          generate(-220,-300,80,-100,340,-300,level,7,15);
          generate(-180,-300,140,-100,400,-300,level,7,15);
        SetFillStyle(1,3);
          SetColor(3);
        for i:=0 to 18 do
```

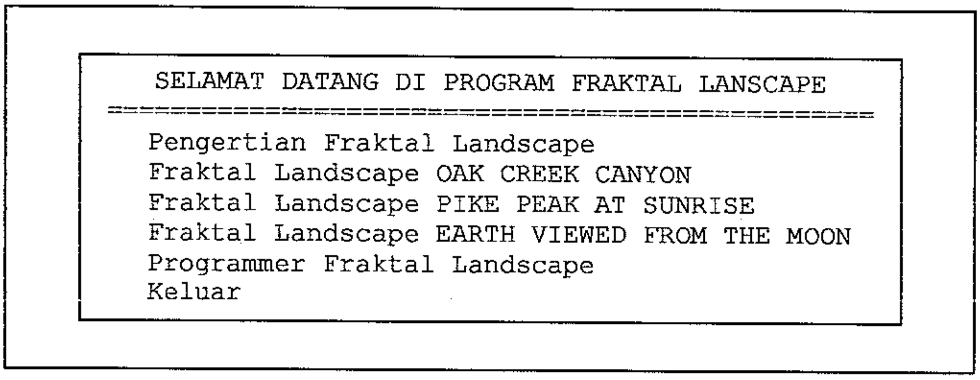
```
                FillEllipse(j[i]+30,k[i],l[i],m[i]);
{Satelit}
          SetFillStyle(2,15);
          SetColor(15);
          Bar(446,100,462,130);
          SetFillStyle(1,12);
          SetColor(12);
          FillEllipse(442,100,8,15);
          SetFillStyle(1,15);
          SetColor(15);
          FillEllipse(440,100,5,15);
          SetLineStyle(0,0,ThickWidth);
          Line(454,100,450,75);
          Line(454,100,458,75);
          SetColor(12);
          Line(440,100,430,100);
          SetColor(15);
          FillEllipse(430,100,2,2);
            ReadKey;Readkey;CloseGraph;
end;

Procedure Tutup;
Var i: integer;
begin
     clrscr;
     GraphDriver:=4;
     GraphMode:=EGAHi;
     InitGraph(GraphDriver,GraphMode,'');
     repeat
          begin
               ClearDevice;
               SetBkColor(0);
               i:=Random(600);
               j:=Random(600);
               l:=l+1;m:=m+1;
               SetColor(Random(15));
               SetLineStyle(SolidLn,0,ThickWidth);
               Circle(i,j,Random(50));
               Circle(10+l,10+l,m);
               Circle(500-l,350-l,m);
               Circle(600-l,l,m);
               Circle(5+l,350-l,m);
               Circle(j,i,Random(50));
               Circle(i+100,j+100,Random(50));
               Circle(j+100,i+50,Random(50));
               if m=100 then m:=0;
               if l=400 then l:=0;
               SetColor(15);
               OutTextXY(200,330,'TERIMA KASIH - TERIMA KASIH -
TERIMA KASIH');
               delay(25);
          end;
     until KeyPressed;
     CloseGraph;
end;


{Program Utama}
Begin
     Clrscr;
     GraphDriver:=4;
     GraphMode:=EGAHi;
     InitGraph(GraphDriver,GraphMode,'');
     repeat
          begin
               ClearDevice;
               SetBkColor(0);
               i:=Random(600);
               j:=Random(600);
               l:=l+1;m:=m+1;
               SetColor(Random(15));
               SetLineStyle(SolidLn,0,ThickWidth);
               Circle(i,j,Random(50));
               Circle(10+l,10+l,m);
               Circle(500-l,350-l,m);
               Circle(600-l,l,m);
               Circle(5+l,350-l,m);
               Circle(j,i,Random(50));
               Circle(i+100,j+100,Random(50));
               Circle(j+100,i+50,Random(50));
               SetBkColor(0);
               if m=100 then m:=0;
               if l=400 then l:=0;
               SetColor(15);
               OutTextXY(300,330,'Tekan Sembarang Tombol Untuk
Melanjutkan');
               delay(50);
          end;
     until KeyPressed;Readkey;
     CloseGraph;
     Repeat
     Key:=menu(0);
     if (key=5) then
     begin
          Tutup;
     end;
     if (key=4) then
     begin
```

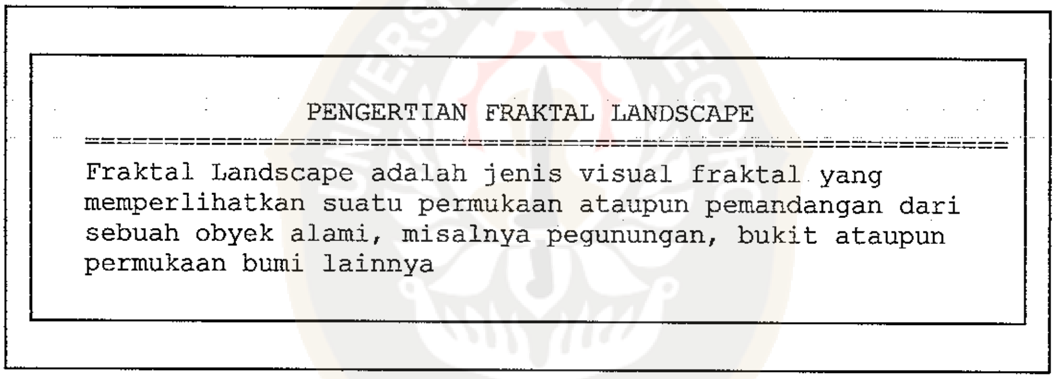
```
          Info;
     end;
          TextBackGround(0);
     if (key=3) then
     begin
          Repeat
          Info;
               if key1<>5 then
               begin
                    level:=key1;
                    Earth;
                    RestoreCrtMode;
               end;
          until key1=5;
          TextBackGround(0);
     end;
     if (key=2) then
     begin
          Repeat
          Info;
               if key1<>5 then
               begin
                    level:=key1;
                    Pike;
                    RestoreCrtMode;
               end;
          until key1=5;
          TextBackGround(0);
     end;
     if (key=1) then
     begin
     repeat
          Info;
               if key1<>5 then
               begin
                    level:=key1;
                    Oak;
                    RestoreCrtMode;
               end;
          until key1=5;
          TextBackGround(0);
     end;
     if (key=0) then
     begin
          Info;
     end;
          TextBackGround(0);
     until key=5;
end.
```

LAMPIRAN 2 : Menu Tampilan Program Fraktal Landscape



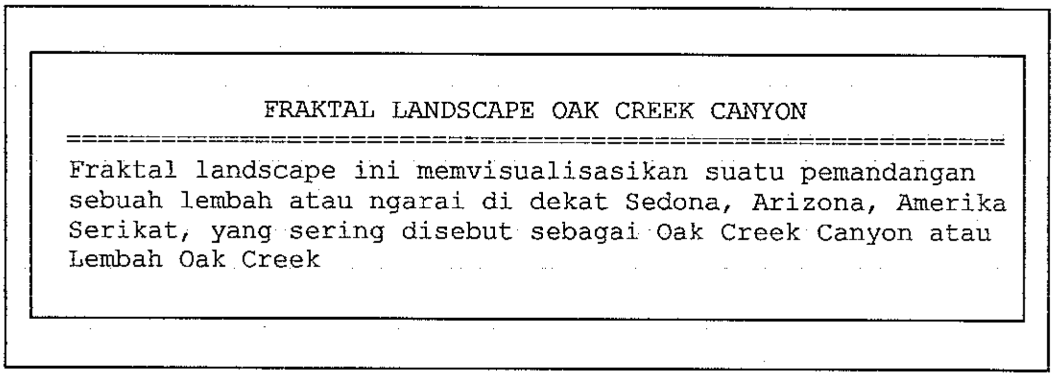
```
   SELAMAT DATANG DI PROGRAM FRAKTAL LANSCAPE
=================================================
  Pengertian Fraktal Landscape
  Fraktal Landscape OAK CREEK CANYON
  Fraktal Landscape PIKE PEAK AT SUNRISE
  Fraktal Landscape EARTH VIEWED FROM THE MOON
  Programmer Fraktal Landscape
  Keluar
```

Tampilan Menu Utama pada Program Fraktal Landscape



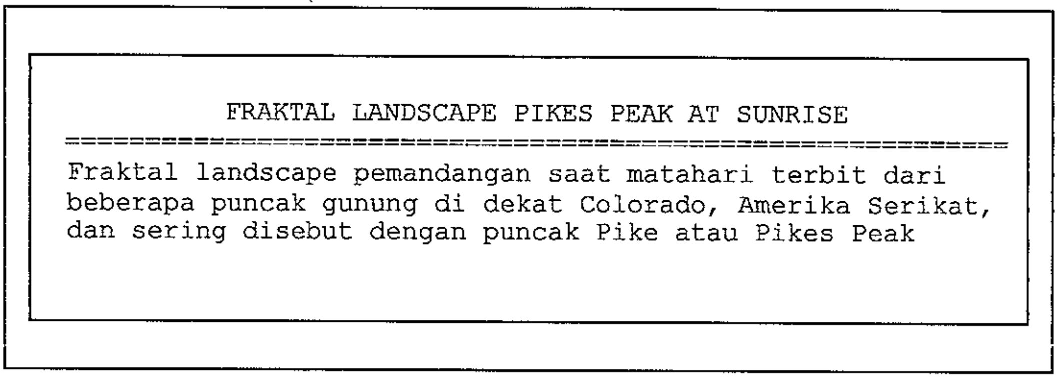
```
             PENGERTIAN FRAKTAL LANDSCAPE
=========================================================
Fraktal Landscape adalah jenis visual fraktal yang
memperlihatkan suatu permukaan ataupun pemandangan dari
sebuah obyek alami, misalnya pegunungan, bukit ataupun
permukaan bumi lainnya
```

Menu Informasi Tentang Pengertian Fraktal Landscape



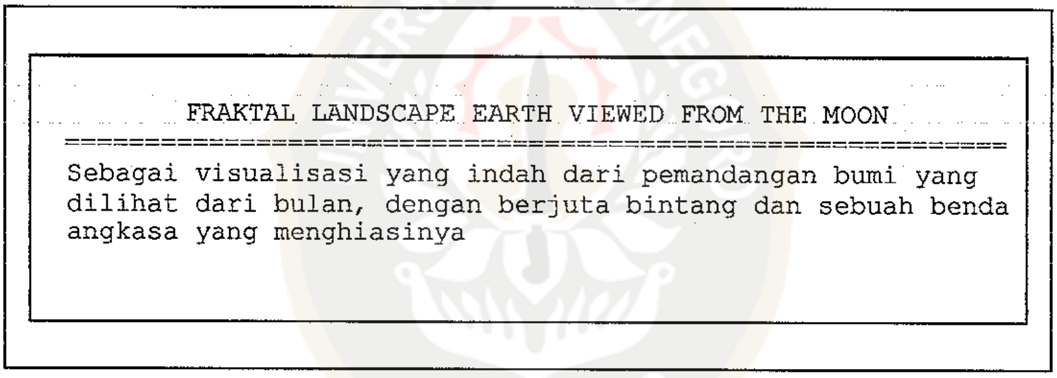
```
            FRAKTAL LANDSCAPE OAK CREEK CANYON
===========================================================
Fraktal landscape ini memvisualisasikan suatu pemandangan
sebuah lembah atau ngarai di dekat Sedona, Arizona, Amerika
Serikat, yang sering disebut sebagai Oak Creek Canyon atau
Lembah Oak Creek
```

Menu Informasi Tentang Fraktal Landscape Oak Creek Canyon



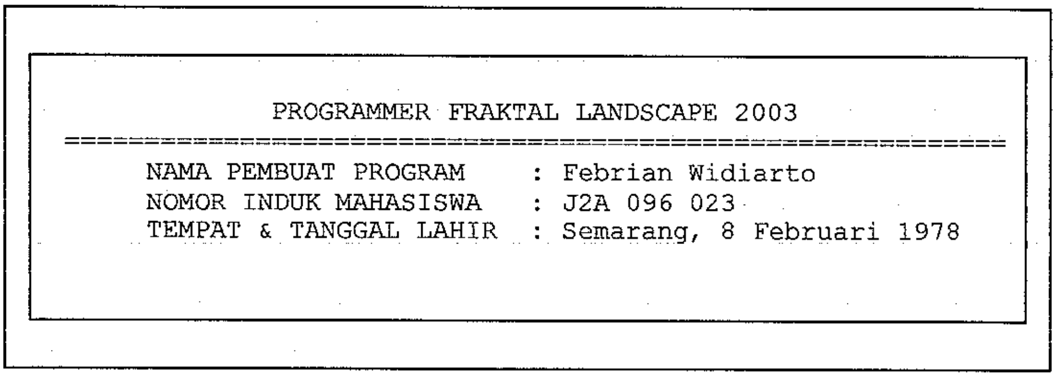
```
          FRAKTAL LANDSCAPE PIKES PEAK AT SUNRISE
==========================================================
Fraktal landscape pemandangan saat matahari terbit dari
beberapa puncak gunung di dekat Colorado, Amerika Serikat,
dan sering disebut dengan puncak Pike atau Pikes Peak
```

Menu Informasi Tentang Fraktal Landscape Pike's Peak At Sunrise



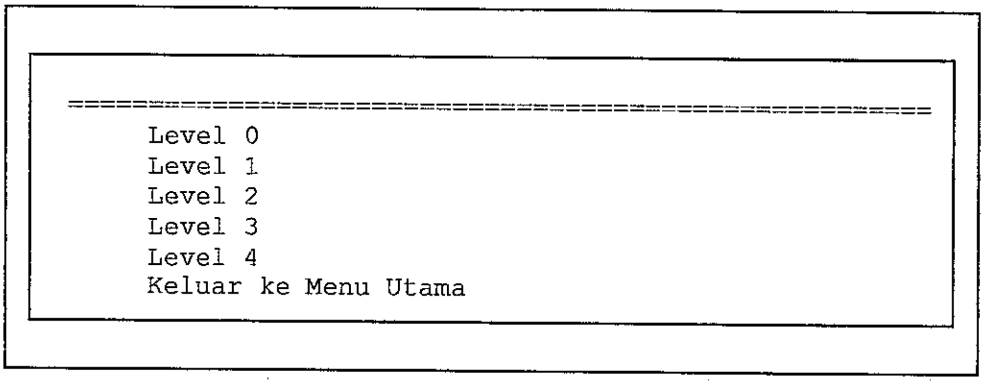
```
       FRAKTAL LANDSCAPE EARTH VIEWED FROM THE MOON
==========================================================
Sebagai visualisasi yang indah dari pemandangan bumi yang
dilihat dari bulan, dengan berjuta bintang dan sebuah benda
angkasa yang menghiasinya
```

Menu Informasi Tentang Fraktal Landscape Earth Viewed From The Moon



```
             PROGRAMMER FRAKTAL LANDSCAPE 2003
==========================================================
     NAMA PEMBUAT PROGRAM     : Febrian Widiarto
     NOMOR INDUK MAHASISWA    : J2A 096 023
     TEMPAT & TANGGAL LAHIR   : Semarang, 8 Februari 1978
```

Menu Informasi Tentang Programmer Fraktal Landscape



```
========================================================
     Level 0
     Level 1
     Level 2
     Level 3
     Level 4
     Keluar ke Menu Utama
```

Gambar Menu Pilihan Level Untuk Menampilkan Fraktal Landscape