LAMPIRAN

Lampiran 1 : Listing Program simulasi Difraksi dengan Borland Delphi 7

```
unit Unit1;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
Forms,
  Dialogs, TeEngine, Series, ExtCtrls, Chart, Grids, StdCtrls, TeeProcs;

  type
    TRGBColor=record
    Red,
    Green,
    Blue:Byte;
    end;

    THSBColor=record
    Hue,
    Saturnation,
    Brightness:Double;
    end;

type
  TForm1 = class(TForm)
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Label4: TLabel;
    Label5: TLabel;
    Label6: TLabel;
    Label7: TLabel;
    Edit1: TEdit;
    Edit2: TEdit;
    Edit3: TEdit;
    Button1: TButton;
    StringGrid1: TStringGrid;
    Button2: TButton;
    Edit4: TEdit;
    Edit5: TEdit;
    Button3: TButton;
    Button4: TButton;
    Chart1: TChart;
    StringGrid2: TStringGrid;
    Button5: TButton;
    StringGrid3: TStringGrid;
    Label9: TLabel;
    Label10: TLabel;
    Label11: TLabel;
    Label12: TLabel;
    Label13: TLabel;
    Label14: TLabel;
    Label15: TLabel;
    Label16: TLabel;
    Series1: TLineSeries;
    procedure Button1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
    procedure Button3Click(Sender: TObject);
    procedure Button4Click(Sender: TObject);
```

```
    procedure Button5Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);

  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;
 JumlahBerkas,JarakAntarPoin,JarakCelahKeLayar,Layar:integer;
 A,B,C,D,E,F,G,H,P,Q,W,X:array[-1000..1000,-1000..1000]of extended;
 i,j,k,l,m,n,r,s: integer;
 BilGelombang,Amplitudo,PanjangGelombang,delta,hasil:double;
implementation

{$R *.dfm}

procedure TForm1.Button1Click(Sender: TObject);
var
ncol:integer;
begin
Label10.Enabled:=true;
Label9.Enabled:=true;
StringGrid1.RowCount:=2;
for ncol:=0 to StringGrid1.ColCount-1 do
StringGrid1.Cols[ncol].Clear;

StringGrid2.RowCount:=2;
for ncol:=0 to StringGrid2.ColCount-1 do
StringGrid2.Cols[ncol].Clear;

StringGrid3.RowCount:=2;
for ncol:=0 to StringGrid3.ColCount-1 do
StringGrid3.Cols[ncol].Clear;

//mulai memasukkan data
  JumlahBerkas := StrToInt(Edit1.Text);
  JarakAntarPoin := StrToInt(Edit2.Text);
  JarakCelahKeLayar := StrToInt(Edit3.Text);
  PanjangGelombang:= StrToFloat(Edit4.Text);
  Amplitudo:= StrToFloat(Edit5.Text);
  BilGelombang:= ((JumlahBerkas*pi)/PanjangGelombang);

   Layar:=JumlahBerkas*20;
  //syarat batas iterasi
  k:= Layar div 2;
  l:= JumlahBerkas div 2;
  //pembedaan ganjil dan genap
  m:= Layar mod 2;
  n:= JumlahBerkas mod 2;
 //mengatur tampilan sringgrid.
 StringGrid1.ColCount:=(2*k)+2;
 StringGrid1.RowCount:=(2*l)+2;
  //memulai iterasi
  for i:=-k to k do
  begin
  StringGrid1.Cells[i+k+1,0]:=IntToStr(i);
        for j:=-l to l do
```

```
        begin
        StringGrid1.Cells[0,j+l+1]:=IntToStr(j);
          if (i=j) then
          A[i,j]:=JarakCelahKeLayar
          else
          A[i,j]:=sqrt(sqr(JarakCelahKeLayar) + sqr(j - i));
          end;
  end;
  //menampilkan ke stringgrid
                for i:=-k to k do
                begin
                        for j:=-l to l do
                        begin

StringGrid1.Cells[i+k+1,j+l+1]:=FloatToStr(A[i,j]);
                        end;
                end;
                StringGrid2.ColCount:=(2*k)+2;
                StringGrid2.RowCount:=(2*l)+2;

                StringGrid3.ColCount:=(2*k)+2;
                StringGrid3.RowCount:=(2*l)+2;
                for i:=-k to k do
                begin           // mencari selisih jarak
                StringGrid2.Cells[i+k+1,0]:=IntToStr(i);
                StringGrid3.Cells[i+k+1,0]:=IntToStr(i);
                  for j:=-l to (l-1) do
                        begin
                     StringGrid2.Cells[0,j+l+1]:=IntToStr(j);
                     StringGrid3.Cells[0,j+l+1]:=IntToStr(j);
                       H[i,j]:=A[i,j+1]-A[i,-l];

StringGrid2.Cells[i+k+1,j+l+1]:=FloatToStr(H[i,j]);
                    P[i,j]:=(BilGelombang*H[i,j]);
                    P[i,j]:=P[i,l-1];
                    StringGrid3.Cells[i+k+1,j+l+1]:=FloatToStr(P[i,j]);
                        end;
                end;

                    end;
procedure TForm1.Button2Click(Sender: TObject);
var
ncol:integer;
begin
Label9.Enabled:=false;
Label10.Enabled:=false;
Label11.Enabled:=true;
Label14.Enabled:=true;
//mengatur tampilan sringgrid.
 StringGrid1.ColCount:=(2*k)+2;
 StringGrid1.RowCount:=(2*l)+2;
 StringGrid2.ColCount:=(2*k)+2;
 StringGrid2.RowCount:=(2*l)+2;
 StringGrid3.RowCount:=(2*k)+2;
 StringGrid3.RowCount:=(2*l)+2;

StringGrid3.RowCount:=2;
for ncol:=0 to StringGrid3.ColCount-1 do
StringGrid3.Cols[ncol].Clear;
```

```
                for i:=-k to k do
                begin
                StringGrid1.Cells[i+k+1,0]:=IntToStr(i);
                StringGrid2.Cells[i+k+1,0]:=IntToStr(i);
                StringGrid3.Cells[i+k+1,0]:=IntToStr(i);
                hasil:=0;
                        for j:=-l to l do
                        begin
                        StringGrid1.Cells[0,j+l+1]:=IntToStr(j);
              B[i,j]:=Amplitudo*cos(BilGelombang*(A[i,j]));
              StringGrid1.Cells[i+k+1,j+l+1]:=FloatToStr(B[i,j]);
                        StringGrid2.Cells[0,j+l+1]:=IntToStr(j);
              E[i,j]:=Amplitudo*sin(BilGelombang*(A[i,j]));
              StringGrid2.Cells[i+k+1,j+l+1]:=FloatToStr(E[i,j]);
                        StringGrid3.Cells[0,j+l+1]:=IntToStr(j);
                        hasil:=hasil+P[i,j];
                        end;
                        Q[i,1]:=hasil;
                        StringGrid3.Cells[i+k+1,1]:=FloatToStr (Q[i,1]);
                        end;
                end;


procedure TForm1.Button3Click(Sender: TObject);
var
nilai1,nilai2,nilai : double;
f,g: integer;
ncol:integer;
begin
Label11.Enabled:=false;
Label14.Enabled:=false;
Label12.Enabled:=true;
Label15.Enabled:=true;
Series1.Clear;
StringGrid1.RowCount:=2;
for ncol:=0 to StringGrid1.ColCount-1 do
StringGrid1.Cols[ncol].Clear;
                for i:=-k to k do
                begin
                StringGrid1.Cells[i+k+1,0]:=IntToStr(i);
                nilai:=0;
                        for j:=-l to l do
                        begin
                        StringGrid1.Cells[0,j+l+1]:=IntToStr(j);
                        nilai:=nilai+B[i,j];
                        end;
                D[i,1]:=nilai;
                StringGrid1.Cells[i+k+1,1]:=FloatToStr(D[i,1]);
                end;
                StringGrid2.RowCount:=2;
                for ncol:=0 to StringGrid2.ColCount-1 do
                StringGrid2.Cols[ncol].Clear;
                 for i:=-k to k do
                begin
                StringGrid2.Cells[i+k+1,0]:=IntToStr(i);
                nilai2:=0;
                        for j:=-l to l do
                        begin
                        StringGrid2.Cells[0,j+l+1]:=IntToStr(j);
                        nilai2:= nilai2 + E[i,j];
```

```
                    end;
                 C[i,1]:=nilai2;
                 StringGrid2.Cells[i+k+1,1]:=FloatToStr(C[i,1]);
                 Series1.AddXY(i,D[i,1]);
               end;
end;

procedure TForm1.Button4Click(Sender: TObject);
var
  nilai,nilai1,delta, intensitas : double;
  Layar,i,JumlahBerkas,lebar,s:integer;
  ncol,max:integer;
  Spectrum: TBitmap;
  r, g, b:byte;
begin
Label13.Enabled:=true;
Label12.Enabled:=false;
Label15.Enabled:=false;
Series1.Clear;
StringGrid1.RowCount:=2;
for ncol:=0 to StringGrid1.ColCount-1 do
StringGrid1.Cols[ncol].Clear;

StringGrid2.RowCount:=2;
for ncol:=0 to StringGrid2.ColCount-1 do
StringGrid2.Cols[ncol].Clear;

StringGrid3.RowCount:=2;
for ncol:=0 to StringGrid3.ColCount-1 do
StringGrid3.Cols[ncol].Clear;

begin
  for i:=-k to k do
    begin
    StringGrid1.Cells[i+k+1,0]:=IntToStr(i);
    delta:=0;
  for j:=-l to l do
    begin
    StringGrid1.Cells[0,j+l+1]:=IntToStr(j);
    intensitas:=
(StrToInt(Edit1.Text))*(sqr(D[i,1])+sqr(C[i,1]))*(sqr(cos(Q[i,1]/2)));
  F[i,1]:=intensitas;
  StringGrid1.Cells[i+k+1,j+l+1]:=FloatToStr(F[i,1]);
  Series1.AddXY(i,F[i,1]);
  end;
end;

procedure TForm1.Button5Click(Sender: TObject);
var
ncol:integer;
begin
Series1.Clear;
for ncol:=0 to StringGrid1.ColCount-1 do
StringGrid1.Cols[ncol].Clear;
for ncol:=0 to StringGrid2.ColCount-1 do
StringGrid2.Cols[ncol].Clear;
for ncol:=0 to StringGrid3.ColCount-1 do
StringGrid3.Cols[ncol].Clear;
end;
procedure TForm1.FormCreate(Sender: TObject);
```

```
begin
Label9.Enabled:=false;
Label10.Enabled:=false;
Label11.Enabled:=false;
Label12.Enabled:=false;
Label13.Enabled:=false;
Label14.Enabled:=false;
Label15.Enabled:=false;
end;
procedure TForm1.Button6Click(Sender: TObject);
var ft,fs:textfile;
  Strs : TStringList;
  i : integer;
begin
  if SaveDialog1.Execute then
    //create a TStringList to save the grid's contents to
    Strs := TStringList.Create;
    try
    //save the row and col counts
    Strs.Add(IntToStr(StringGrid1.RowCount));
    Strs.Add(IntToStr(StringGrid1.ColCount));

    //save each rows content as comma delimited text
    if StringGrid1.RowCount > 0 then begin
    for i := 0 to StringGrid1.RowCount -1 do
    Strs.Add( StringGrid1.Rows[i].CommaText );
    end;
    //save the stringlist to the file
    Strs.SaveToFile(SaveDialog1.FileName);
    finally
    Strs.Free;
    end;
    end;
procedure TForm1.Button7Click(Sender: TObject);
var ft,fs:textfile;
  Strs : TStringList;
  i : integer;
begin
  if SaveDialog2.Execute then
    //create a TStringList to save the grid's contents to
    Strs := TStringList.Create;
    try
    //save the row and col counts
    Strs.Add(IntToStr(StringGrid2.RowCount));
    Strs.Add(IntToStr(StringGrid2.ColCount));

    //save each rows content as comma delimited text
    if StringGrid2.RowCount > 0 then begin
    for i := 0 to StringGrid2.RowCount -1 do
    Strs.Add( StringGrid2.Rows[i].CommaText );
    end;

    //save the stringlist to the file
    Strs.SaveToFile(SaveDialog2.FileName);
    finally
    Strs.Free;
    end;
    end;

procedure TForm1.Button8Click(Sender: TObject);
```

```
var ft,fs:textfile;
  Strs : TStringList;
  i : integer;
begin
  if SaveDialog3.Execute then
    //create a TStringList to save the grid's contents to
    Strs := TStringList.Create;
    try
    //save the row and col counts
    Strs.Add(IntToStr(StringGrid3.RowCount));
    Strs.Add(IntToStr(StringGrid3.ColCount));

    //save each rows content as comma delimited text
    if StringGrid3.RowCount > 0 then begin
    for i := 0 to StringGrid3.RowCount -1 do
    Strs.Add( StringGrid3.Rows[i].CommaText );
    end;

    //save the stringlist to the file
    Strs.SaveToFile(SaveDialog3.FileName);
    finally
    Strs.Free;
    end;
    end;
end.
```

Lampiran 2 : Data Hasil Analisis Difraksi dengan Microsoft Excel 2007

Jumlah Berkas	3
Point di Layar	5
Jarak celah-layar	4
Panjang Gelomb	6
Amplitudo	2
Intensitas Maks	10
Bil Gelomb (k)	1,570796
pi	3,141593

			Pitagoras							
	-30	-29	-28	-27	-26	-25	-24	-23	-22	-21
-1	29,27456	28,28427	27,29469	26,30589	25,31798	24,33105	23,34524	22,36068	21,37756	20,39608
0	30,26549	29,27456	28,28427	27,29469	26,30589	25,31798	24,33105	23,34524	22,36068	21,37756
1	31,257	30,26549	29,27456	28,28427	27,29469	26,30589	25,31798	24,33105	23,34524	22,36068
-20	-19	-18	-17	-16	-15	-14	-13	-12	-11	
19,41649	18,43909	17,46425	16,49242	15,52417	14,56022	13,60147	12,64911	11,7047	10,77033	
20,39608	19,41649	18,43909	17,46425	16,49242	15,52417	14,56022	13,60147	12,64911	11,7047	
21,37756	20,39608	19,41649	18,43909	17,46425	16,49242	15,52417	14,56022	13,60147	12,64911	
-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	
9,848858	8,944272	8,062258	7,211103	6,403124	5,656854	5	4,472136	4,123106	4	
10,77033	9,848858	8,944272	8,062258	7,211103	6,403124	5,656854	5	4,472136	4,123106	
11,7047	10,77033	9,848858	8,944272	8,062258	7,211103	6,403124	5,656854	5	4,472136	
0	1	2	3	4	5	6	7	8	9	10
4,123106	4,472136	5	5,656854	6,403124	7,211103	8,062258	8,944272	9,848858	10,77033	11,7047
4	4,123106	4,472136	5	5,656854	6,403124	7,211103	8,062258	8,944272	9,848858	10,77033
4,123106	4	4,123106	4,472136	5	5,656854	6,403124	7,211103	8,062258	8,944272	9,848858
11	12	13	14	15	16	17	18	19	20	
12,64911	13,60147	14,56022	15,52417	16,49242	17,46425	18,43909	19,41649	20,39608	21,37756	
11,7047	12,64911	13,60147	14,56022	15,52417	16,49242	17,46425	18,43909	19,41649	20,39608	
10,77033	11,7047	12,64911	13,60147	14,56022	15,52417	16,49242	17,46425	18,43909	19,41649	
21	22	23	24	25	26	27	28	29	30	
22,36068	23,34524	24,33105	25,31798	26,30589	27,29469	28,28427	29,27456	30,26549	31,257	
21,37756	22,36068	23,34524	24,33105	25,31798	26,30589	27,29469	28,28427	29,27456	30,26549	
20,39608	21,37756	22,36068	23,34524	24,33105	25,31798	26,30589	27,29469	28,28427	29,27456	

			Gelombang Real							
	-30	-29	-28	-27	-26	-25	-24	-23	-22	-21
-1	-0,83607	1,8039	0,893081	-1,77353	-0,95794	1,735626	1,032204	-1,68751	-1,11781	1,625245
0	-1,82859	-0,83607	1,8039	0,893081	-1,77353	-0,95794	1,735626	1,032204	-1,68751	-1,11781
1	0,785635	-1,82859	-0,83607	1,8039	0,893081	-1,77353	-0,95794	1,735626	1,032204	-1,68751
-20	-19	-18	-17	-16	-15	-14	-13	-12	-11	-10
1,217077	-1,54285	-1,33261	1,430946	1,466884	-1,27432	-1,62075	1,047378	1,788668	-0,70598	-1,9439
1,625245	1,217077	-1,54285	-1,33261	1,430946	1,466884	-1,27432	-1,62075	1,047378	1,788668	-0,70598
-1,11781	1,625245	1,217077	-1,54285	-1,33261	1,430946	1,466884	-1,27432	-1,62075	1,047378	1,788668
-9	-8	-7	-6	-5	-4	-3	-2	-1	0	
0,174851	1,990444	0,651111	-1,61225	-1,71643	-2,1E-09	1,474738	1,962723	2	1,962723	
-1,9439	0,174851	1,990444	0,651111	-1,61225	-1,71643	-2,1E-09	1,474738	1,962723	2	
-0,70598	-1,9439	0,174851	1,990444	0,651111	-1,61225	-1,71643	-2,1E-09	1,474738	1,962723	
1	2	3	4	5	6	7	8	9	10	
1,474738	-2,1E-09	-1,71643	-1,61225	0,651111	1,990444	0,174851	-1,9439	-0,70598	1,788668	
1,962723	1,474738	-2,1E-09	-1,71643	-1,61225	0,651111	1,990444	0,174851	-1,9439	-0,70598	
2	1,962723	1,474738	-2,1E-09	-1,71643	-1,61225	0,651111	1,990444	0,174851	-1,9439	
11	12	13	14	15	16	17	18	19	20	
1,047378	-1,62075	-1,27432	1,466884	1,430946	-1,33261	-1,54285	1,217077	1,625245	-1,11781	
1,788668	1,047378	-1,62075	-1,27432	1,466884	1,430946	-1,33261	-1,54285	1,217077	1,625245	
-0,70598	1,788668	1,047378	-1,62075	-1,27432	1,466884	1,430946	-1,33261	-1,54285	1,217077	
21	22	23	24	25	26	27	28	29	30	
-1,68751	1,032204	1,735626	-0,95794	-1,77353	0,893081	1,8039	-0,83607	-1,82859	0,785635	
-1,11781	-1,68751	1,032204	1,735626	-0,95794	-1,77353	0,893081	1,8039	-0,83607	-1,82859	
1,625245	-1,11781	-1,68751	1,032204	1,735626	-0,95794	-1,77353	0,893081	1,8039	-0,83607	

			Gelombang Imaginer							
	-30	-29	-28	-27	-26	-25	-24	-23	-22	-21
-1	1,816862	0,863681	-1,78953	-0,92444	1,755665	0,993782	-1,71305	-1,07346	1,658461	1,16558
0	-0,8101	1,816862	0,863681	-1,78953	-0,92444	1,755665	0,993782	-1,71305	-1,07346	1,658461
1	-1,83923	-0,8101	1,816862	0,863681	-1,78953	-0,92444	1,755665	0,993782	-1,71305	-1,07346
-20	-19	-18	-17	-16	-15	-14	-13	-12	-11	
-1,58705	-1,27264	1,491361	1,397281	-1,3595	-1,54147	1,17183	1,703819	-0,8948	-1,87125	
1,16558	-1,58705	-1,27264	1,491361	1,397281	-1,3595	-1,54147	1,17183	1,703819	-0,8948	
1,658461	1,16558	-1,58705	-1,27264	1,491361	1,397281	-1,3595	-1,54147	1,17183	1,703819	
-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0
0,470379	1,992342	0,195277	-1,89105	-1,1835	1,026577	2	1,350981	0,384342	1,64E-09	0,384342
-1,87125	0,470379	1,992342	0,195277	-1,89105	-1,1835	1,026577	2	1,350981	0,384342	1,64E-09
-0,8948	-1,87125	0,470379	1,992342	0,195277	-1,89105	-1,1835	1,026577	2	1,350981	0,384342
1	2	3	4	5	6	7	8	9	10	
1,350981	2	1,026577	-1,1835	-1,89105	0,195277	1,992342	0,470379	-1,87125	-0,8948	
0,384342	1,350981	2	1,026577	-1,1835	-1,89105	0,195277	1,992342	0,470379	-1,87125	
1,64E-09	0,384342	1,350981	2	1,026577	-1,1835	-1,89105	0,195277	1,992342	0,470379	
11	12	13	14	15	16	17	18	19	20	
1,703819	1,17183	-1,54147	-1,3595	1,397281	1,491361	-1,27264	-1,58705	1,16558	1,658461	
-0,8948	1,703819	1,17183	-1,54147	-1,3595	1,397281	1,491361	-1,27264	-1,58705	1,16558	
-1,87125	-0,8948	1,703819	1,17183	-1,54147	-1,3595	1,397281	1,491361	-1,27264	-1,58705	
21	22	23	24	25	26	27	28	29	30	
-1,07346	-1,71305	0,993782	1,755665	-0,92444	-1,78953	0,863681	1,816862	-0,8101	-1,83923	
1,658461	-1,07346	-1,71305	0,993782	1,755665	-0,92444	-1,78953	0,863681	1,816862	-0,8101	
1,16558	1,658461	-1,07346	-1,71305	0,993782	1,755665	-0,92444	-1,78953	0,863681	1,816862	

Selisih Jarak

	-30	-29	-28	-27	-26	-25	-24	-23	-22	-21
-1	0,99093	0,990291	0,989583	0,988795	0,987915	0,986928	0,985815	0,984555	0,983121	0,98148
0	1,982437	1,981221	1,979874	1,978378	1,97671	1,974843	1,972743	1,97037	1,967677	1,964602
-20	-19	-18	-17	-16	-15	-14	-13	-12	-11	
0,97959	0,977399	0,97484	0,971827	0,968248	0,963955	0,958749	0,95236	0,944411	0,93437	
1,96107	1,956989	1,952239	1,946666	1,940075	1,932203	1,922704	1,911109	1,896771	1,878781	
-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0
0,921472	0,904586	0,882014	0,851155	0,807978	0,74627	0,656854	0,527864	0,34903	0,123106	-0,12311
1,855842	1,826058	1,7866	1,733169	1,659134	1,554248	1,403124	1,184718	0,876894	0,472136	0
1	2	3	4	5	6	7	8	9	10	
-0,34903	-0,52786	-0,65685	-0,74627	-0,80798	-0,85116	-0,88201	-0,90459	-0,92147	-0,93437	
-0,47214	-0,87689	-1,18472	-1,40312	-1,55425	-1,65913	-1,73317	-1,7866	-1,82606	-1,85584	
11	12	13	14	15	16	17	18	19	20	
-0,94441	-0,95236	-0,95875	-0,96395	-0,96825	-0,97183	-0,97484	-0,9774	-0,97959	-0,98148	
-1,87878	-1,89677	-1,91111	-1,9227	-1,9322	-1,94007	-1,94667	-1,95224	-1,95699	-1,96107	
21	22	23	24	25	26	27	28	29	30	
-0,98312	-0,98456	-0,98582	-0,98693	-0,98792	-0,9888	-0,98958	-0,99029	-0,99093	-0,99151	
-1,9646	-1,96768	-1,97037	-1,97274	-1,97484	-1,97671	-1,97838	-1,97987	-1,98122	-1,98244	

Delta

	-30	-29	-28	-27	-26	-25	-24	-23	-22	-21
-1	1,556549	1,555546	1,554434	1,553196	1,551813	1,550262	1,548515	1,546536	1,544284	1,541706
0	3,114005	3,112094	3,109979	3,107629	3,105009	3,102076	3,098777	3,095051	3,090819	3,085989
-20	-19	-18	-17	-16	-15	-14	-13	-12	-11	
1,538737	1,535295	1,531275	1,526542	1,52092	1,514177	1,506	1,495963	1,483477	1,467705	
3,080442	3,074031	3,066569	3,057816	3,047462	3,035097	3,020177	3,001963	2,97944	2,951182	
-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0
1,447445	1,42092	1,385465	1,336991	1,269169	1,172238	1,031784	0,829167	0,548256	0,193374	-0,19337
2,91515	2,868365	2,806385	2,722456	2,606161	2,441408	2,204022	1,860951	1,377422	0,741629	0
1	2	3	4	5	6	7	8	9	10	
-0,54826	-0,82917	-1,03178	-1,17224	-1,26917	-1,33699	-1,38546	-1,42092	-1,44744	-1,46771	
-0,74163	-1,37742	-1,86095	-2,20402	-2,44141	-2,60616	-2,72246	-2,80638	-2,86836	-2,91515	
11	12	13	14	15	16	17	18	19	20	
-1,48348	-1,49596	-1,506	-1,51418	-1,52092	-1,52654	-1,53127	-1,53529	-1,53874	-1,54171	
-2,95118	-2,97944	-3,00196	-3,02018	-3,0351	-3,04746	-3,05782	-3,06657	-3,07403	-3,08044	
21	22	23	24	25	26	27	28	29	30	
-1,54428	-1,54654	-1,54851	-1,55026	-1,55181	-1,5532	-1,55443	-1,55555	-1,55655	-1,55746	
-3,08599	-3,09082	-3,09505	-3,09878	-3,10208	-3,10501	-3,10763	-3,10998	-3,11209	-3,114	

Superposisi Real

point	-30	-29	-28	-27	-26	-25	-24	-23	-22	-21
superposi	-1,87902	-0,86076	1,86091	0,923449	-1,83839	-0,99584	1,809895	1,08032	-1,77312	-1,18008
-20	-19	-18	-17	-16	-15	-14	-13	-12	-11	
1,724508	1,299473	-1,65838	-1,44451	1,565222	1,623514	-1,42818	-1,84768	1,215301	2,130065	
-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0
-0,86121	-2,47503	0,221396	2,816406	1,029309	-2,67757	-3,32868	-0,24169	3,437461	5,437461	5,925446
1	2	3	4	5	6	7	8	9	10	
5,437461	3,437461	-0,24169	-3,32868	-2,67757	1,029309	2,816406	0,221396	-2,47503	-0,86121	
11	12	13	14	15	16	17	18	19	20	
2,130065	1,215301	-1,84768	-1,42818	1,623514	1,565222	-1,44451	-1,65838	1,299473	1,724508	

21	22	23	24	25	26	27	28	29	30
-1,18008	-1,77312	1,08032	1,809895	-0,99584	-1,83839	0,923449	1,86091	-0,86076	-1,87902

			Superposisi Imaginer						
point	-30	-29	-28	-27	-26	-25	-24	-23	-22	-21
superposi	-0,83247	1,870442	0,891015	-1,85028	-0,9583	1,82501	1,036393	-1,79273	-1,12805	1,750585
-20	-19	-18	-17	-16	-15	-14	-13	-12	-11
1,236992	-1,69411	-1,36833	1,616	1,529137	-1,50369	-1,72914	1,334182	1,980848	-1,06224
-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0
-2,29568	0,591467	2,657998	0,296573	-2,87927	-2,04797	1,84308	4,377557	3,735323	1,735323	0,768684
1	2	3	4	5	6	7	8	9	10
1,735323	3,735323	4,377557	1,84308	-2,04797	-2,87927	0,296573	2,657998	0,591467	-2,29568
11	12	13	14	15	16	17	18	19	20
-1,06224	1,980848	1,334182	-1,72914	-1,50369	1,529137	1,616	-1,36833	-1,69411	1,236992
21	22	23	24	25	26	27	28	29	30
1,750585	-1,12805	-1,79273	1,036393	1,82501	-0,9583	-1,85028	0,891015	1,870442	-0,83247

			Intensitas						
point	-30	-29	-28	-27	-26	-25	-24	-23	-22	-21
intensitas	0,002411	0,002767	0,003191	0,003699	0,004314	0,005062	0,00598	0,007116	0,008537	0,010333
-20	-19	-18	-17	-16	-15	-14	-13	-12	-11
0,012628	0,0156	0,019504	0,024715	0,031796	0,041614	0,055541	0,075824	0,106269	0,153592
-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0
0,230211	0,36032	0,593925	1,041322	1,962748	4,010367	8,865411	20,58325	46,08117	84,89824	107,1054
1	2	3	4	5	6	7	8	9	10
84,89824	46,08117	20,58325	8,865411	4,010367	1,962748	1,041322	0,593925	0,36032	0,230211
11	12	13	14	15	16	17	18	19	20
0,153592	0,106269	0,075824	0,055541	0,041614	0,031796	0,024715	0,019504	0,0156	0,012628
21	22	23	24	25	26	27	28	29	30
0,010333	0,008537	0,007116	0,00598	0,005062	0,004314	0,003699	0,003191	0,002767	0,002411

Lampiran 3 : Data Intensitas

Uji awal

Jumlah Berkas	3
Point di Layar	5
Jarak celah-layar	4
Panjang Gelombang	6
Amplitudo	2

Coordinate

```
2
62
,-30,-29,-28,-27,-26,-25,-24,-23,-22,-21,-20,-19,-18,-17,-16,-15,-14,-
13,-12,-11,-10,-9,-8,-7,-6,-5,-4,-3,-2,-
1,0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,
27,28,29,30
-
```

Value

```
1,"0,00241087226161777","0,00276656594973365","0,00319053641332002","0,00
369917458767882","0,0043136622288821","0,00506164424357783","0,0059795882
3974974","0,00711616011021044","0,00853712606834326","0,0103325881624135"
,"0,0126278549351322","0,0156000926679679","0,0195043778332634","0,024715
4205553541","0,0317961277571682","0,0416135288298121","0,0555410868737199
","0,0758244455114414","0,10626919831476","0,153592352368483","0,23021129
9901736","0,360319637445929","0,593924528008526","1,04132175095436","1,96
274794696091","4,01036705795728","8,86541061580999","20,5832462891119","4
6,0811694312568","84,8982404028728","107,105350221431","84,8982404028728"
,"46,0811694312568","20,5832462891119","8,86541061580999","4,010367057957
28","1,96274794696091","1,04132175095436","0,593924528008526","0,36031963
7445929","0,230211299901736","0,153592352368483","0,10626919831476","0,07
58244455114414","0,0555410868737199","0,0416135288298121","0,031796127757
1682","0,0247154205553541","0,0195043778332634","0,0156000926679679","0,0
126278549351322","0,0103325881624135","0,00853712606834326","0,0071161601
1021044","0,00597958823974974","0,00506164424357783","0,0043136622288821"
,"0,00369917458767882","0,00319053641332002","0,00276656594973365","0,002
41087226161777"
```

Input 1

Jumlah Berkas	30
Point di Layar	30
Jarak celah-layar	100

Panjang Gelombang	405
Amplitudo	1

Coordinate

,-300,-299,-298,-297,-296,-295,-294,-293,-292,-291,-290,-289,-288,-287,-286,-285,-284,-283,-282,-281,-280,-279,-278,-277,-276,-275,-274,-273,-272,-271,-270,-269,-268,-267,-266,-265,-264,-263,-262,-261,-260,-259,-258,-257,-256,-255,-254,-253,-252,-251,-250,-249,-248,-247,-246,-245,-244,-243,-242,-241,-240,-239,-238,-237,-236,-235,-234,-233,-232,-231,-230,-229,-228,-227,-226,-225,-224,-223,-222,-221,-220,-219,-218,-217,-216,-215,-214,-213,-212,-211,-210,-209,-208,-207,-206,-205,-204,-203,-202,-201,-200,-199,-198,-197,-196,-195,-194,-193,-192,-191,-190,-189,-188,-187,-186,-185,-184,-183,-182,-181,-180,-179,-178,-177,-176,-175,-174,-173,-172,-171,-170,-169,-168,-167,-166,-165,-164,-163,-162,-161,-160,-159,-158,-157,-156,-155,-154,-153,-152,-151,-150,-149,-148,-147,-146,-145,-144,-143,-142,-141,-140,-139,-138,-137,-136,-135,-134,-133,-132,-131,-130,-129,-128,-127,-126,-125,-124,-123,-122,-121,-120,-119,-118,-117,-116,-115,-114,-113,-112,-111,-110,-109,-108,-107,-106,-105,-104,-103,-102,-101,-100,-99,-98,-97,-96,-95,-94,-93,-92,-91,-90,-89,-88,-87,-86,-85,-84,-83,-82,-81,-80,-79,-78,-77,-76,-75,-74,-73,-72,-71,-70,-69,-68,-67,-66,-65,-64,-63,-62,-61,-60,-59,-58,-57,-56,-55,-54,-53,-52,-51,-50,-49,-48,-47,-46,-45,-44,-43,-42,-41,-40,-39,-38,-37,-36,-35,-34,-33,-32,-31,-30,-29,-28,-27,-26,-25,-24,-23,-22,-21,-20,-19,-18,-17,-16,-15,-14,-13,-12,-11,-10,-9,-8,-7,-6,-5,-4,-3,-2,-1,0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52,53,54,55,56,57,58,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,76,77,78,79,80,81,82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,97,98,99,100,101,102,103,104,105,106,107,108,109,110,111,112,113,114,115,116,117,118,119,120,121,122,123,124,125,126,127,128,129,130,131,132,133,134,135,136,137,138,139,140,141,142,143,144,145,146,147,148,149,150,151,152,153,154,155,156,157,158,159,160,161,162,163,164,165,166,167,168,169,170,171,172,173,174,175,176,177,178,179,180,181,182,183,184,185,186,187,188,189,190,191,192,193,194,195,196,197,198,199,200,201,202,203,204,205,206,207,208,209,210,211,212,213,214,215,216,217,218,219,220,221,222,223,224,225,226,227,228,229,230,231,232,233,234,235,236,237,238,239,240,241,242,243,244,245,246,247,248,249,250,251,252,253,254,255,256,257,258,259,260,261,262,263,264,265,266,267,268,269,270,271,272,273,274,275,276,277,278,279,280,281,282,283,284,285,286,287,288,289,290,291,292,293,294,295,296,297,298,299,300

Value

15,"183,775157316717","182,505636338764","181,22673205878","179,938397392674","178,640586429825","177,333254508751","176,016358295988","174,68985586832","173,353706798483","172,007872244485","170,65231504267","169,286999804701","167,911893018579","166,526963153891","165,132180771427","163,727518637352","162,312951842105","160,888457924208","159,454016999184","158,009611893773","156,555228285664","155,090854848946","153,616483405513","152,132109082638","150,637730476978","149,133349825232","147,61897318174","146,094610603266","144,560276341254","143,015989041851","141,461771953985","139,897653145818","138,32366572989","136,739848097296","135,146244161227","133,542903610262","131,929882171752","130,307241885707","128,675051389574","127,033386214324","125,382329092283","123,721970277147","122,052407876644","120,373748198325","118,68610610898","116,989605408185","115,284379216523","113,570570379017","111,84833188436","110,117827300508","108,379231227277","106,632729766543","104,878521010719","103,116815550175","101,347837000291","99,5718225488816","97,7890235247138","95,999705987897","94,2041513429392","92,4026569752784","90,5955369121402","88,7831225085847","86,965763159643","85,1438270394609","83,3177018684061","81,4877957091079","79,6545377924441","77,8183793745","75,9797946255671","74,1392815

52267","72,2973629539221","70,4545874143163","68,6115303300208","66,76879
49764878","64,9270136131382","63,0868486287008","61,2489937280806","59,41
41751620684","57,583153001211","55,7567224552026","53,9357152391637","52,
1210009882012","50,3134887216549","48,5141283584553","46,7239122850217","
44,9438769771451","43,1751046773006","41,4187251288345","39,675917368471"
,"37,9479115785698","36,2359910005537","34,5414939109065","32,86581566110
59","31,2104107828319","29,5767951597352","27,9665482670056","26,38131547
99117","24,8228104524093","23,2928175668263","21,7931944555353","20,32587
45954002","18,8928699756581","17,4962738397409","16,138263501373","14,821
1032350891","13,5471472410979","12,3188426841793","11,1387328060333","10,
0094601101976","8,93376961832252","7,91451219622273","6,95464794772071","
6,05724967385008","5,22550639449488","4,46272692900023","3,77234353169669
","3,1579155776321","2,6231332930922","2,17182152471557","1,8079435401612
","1,53560485236212","1,3590570583936","1,28270168288989","1,311094014755
37","1,44894692462635","1,70113464914153","2,07269652656506","2,568840666
66842","3,19494753600621","3,95657343780963","4,85945386366005","5,909506
6918817","7,11283520520202","8,47573089765483","10,0046760379385","11,706
3459534754","13,5876109962402","15,6555381480187","17,9173922191193","20,
3806365906673","23,0529334464611","25,9421434359461","29,0563247051508","
32,4037312274297","35,9928103605386","39,8321995509504","43,930722100351"
,"48,2973819029819","52,9413570558552","57,8719922368944","63,09878973874
07","68,6313990382824","74,4796047739608","80,6533129945537","87,16253553
44601","94,0173723615328","101,227991734232","108,804607995355","116,7574
56819847","125,09676772431","133,832733635769","142,97547730721","152,535
014357366","162,521212702321","172,943748136863","183,812055814227","195,
135277364156","206,922203381171","219,181211007798","231,920196331591","2
45,146501310152","258,866834935548","273,087188348671","287,812743615674"
,"303,047775883093","318,795548635959","335,058201794793","351,8366324034
","369,130367680397","386,937430234218","405,254195274733","424,075239695
329","443,393182948369","463,198519695322","483,479444281628","504,221667
166682","525,408223532491","547,019274401807","569,03190071939","591,4198
90989883","614,153523224202","637,199342124925","660,519932641571","684,0
73691250522","707,81459656343","731,691981143772","755,650306715538","779
,628945282247","803,56196903997","827,377952366068","850,999789596712","8
74,344532771637","897,323254024112","919,840937827547","941,796408876405"
,"963,082301976676","983,585080947406","1003,18511418619","1021,756815223
41","1039,16885727631","1055,28447150701","1069,96183937943","1083,054590
18707","1094,41241547366","1103,88181267661","1111,30697086996","1116,530
81194938","1119,39620096197","1119,74733951268","1117,431356246","1112,30
010827459","1104,21220706835","1093,03528169058","1078,64849132848","1060
,94529777089","1039,83650678913","1015,25358522931","987,152257978907","9
55,51638577724","920,362121053516","881,742334552785","839,751300411531",
"794,529621538208","746,269370617259","695,219414776513","641,69088394019
","586,062734153467","528,787347750093","470,396102206352","411,504828974
199","352,819072634942","295,139049517634","239,364193673929","186,497167
022605","137,647199840991","94,0326178957963","56,9824037209863","27,9366
322514663","8,445615628767 38","0,167588957865101","4,864768583091","24,39
76175480538","60,7171597914951","115,855195775396","191,912288079492","29
1,04340641355","415,441147810136","567,316479687886","748,876991123095","
962,302681000754","1209,71936053602","1493,16980158055","1814,58282056864
","2175,7405501113","2578,24421508141","3023,47879628852","3512,577031032
89","4046,38326425988","4625,41772482705","5249,84185651667","5919,425380
74071","6633,51580521045","7391,01111801609","8190,33641751344","9029,425
22325474","9905,7061903","10816,0959073302","11756,9983972144","12724,311
8567378","13713,4430703465","14719,3298118962","15736,4714101062","16758,
9674999544","17780,5648165654","18794,7117137873","19794,6199107782","207
73,3327911023","21723,7994050154","22638,9531619181","23511,7940515106","
24335,4731039829","25103,3776962549","25809,2162369354","26447,100720701"
,"27011,6256357279","27497,941737167","27901,8232658865","28219,727294087
3","28448,84401606","28587,1369702826","28633,3723741887","28587,13697028
26","28448,84401606","28219,7272940873","27901,8232658865","27497,9417371
67","27011,6256357279","26447,100720701","25809,2162369354","25103,377696
2549","24335,4731039829","23511,7940515106","22638,9531619181","21723,799

4050154","20773,3327911023","19794,6199107782","18794,7117137873","17780,
5648165654","16758,9674999544","15736,4714101062","14719,3298118963","137
13,4430703465","12724,3118567378","11756,9983972144","10816,0959073303","
9905,7061903","9029,42522325474","8190,33641751344","7391,01111801609","6
633,51580521045","5919,42538074071","5249,84185651667","4625,41772482705"
,"4046,38326425988","3512,57703103289","3023,47879628852","2578,244215081
41","2175,7405501113","1814,58282056864","1493,16980158055","1209,7193605
3602","962,302681000754","748,876991123095","567,316479687885","415,44114
7810136","291,04340641355","191,912288079492","115,855195775396","60,7171
597914952","24,3976175480538","4,864768583091","0,167588957865101","8,445
61562876738","27,9366322514663","56,9824037209862","94,0326178957963","13
7,647199840991","186,497167022605","239,364193673929","295,139049517634",
"352,819072634942","411,5048289742","470,396102206352","528,787347750093"
,"586,062734153467","641,69088394019","695,219414776513","746,26937061725
9","794,529621538208","839,751300411532","881,742334552785","920,36212105
3516","955,51638577724","987,152257978908","1015,25358522931","1039,83650
678913","1060,94529777089","1078,64849132848","1093,03528169058","1104,21
220706835","1112,30010827459","1117,43135624601","1119,74733951268","1119
,39620096197","1116,53081194938","1111,30697086996","1103,88181267661","1
094,41241547366","1083,05459018707","1069,96183937943","1055,28447150701"
,"1039,16885727631","1021,75681522341","1003,18511418619","983,5850809474
07","963,082301976677","941,796408876406","919,840937827548","897,3232540
24113","874,344532771637","850,999789596713","827,377952366068","803,5619
6903997","779,628945282247","755,650306715538","731,691981143772","707,81
459656343","684,073691250522","660,51993264157","637,199342124925","614,1
53523224202","591,419890989883","569,031900719391","547,019274401808","52
5,408223532491","504,221667166682","483,479444281628","463,198519695322",
"443,393182948369","424,07523969533","405,254195274734","386,937430234219
","369,130367680398","351,836632403401","335,058201794793","318,795548635
96","303,047775883094","287,812743615673","273,08718834867","258,86683493
5548","245,146501310152","231,920196331591","219,181211007798","206,92220
3381171","195,135277364157","183,812055814227","172,943748136863","162,52
1212702321","152,535014357365","142,975477307209","133,832733635769","125
,09676772431","116,757456819847","108,804607995355","101,227991734232","9
4,0173723615328","87,1625355344601","80,6533129945537","74,479604773961",
"68,6313990382823","63,0987897387407","57,8719922368943","52,941357055855
2","48,2973819029821","43,9307221003509","39,8321995509504","35,992810360
5388","32,4037312274297","29,0563247051508","25,9421434359461","23,052933
4464611","20,3806365906673","17,9173922191193","15,6555381480187","13,587
6109962402","11,7063459534754","10,0046760379385","8,47573089765482","7,1
1283520520201","5,90950669188168","4,85945386366004","3,95657343780962","
3,19494753600618","2,56884066666842","2,07269652656506","1,70113464914152
","1,44894692462638","1,31109401475538","1,28270168288989","1,35905705839
359","1,5356048523621","1,8079435401612","2,17182152471558","2,6231332930
922","3,15791557763211","3,77234353169669","4,46272692900022","5,22550639
449493","6,0572496738501","6,95464794772069","7,91451219622272","8,933769
61832256","10,0094601101976","11,1387328060333","12,3188426841793","13,54
71472410979","14,8211032350892","16,1382635013731","17,4962738397409","18
,8928699756582","20,3258745954002","21,7931944555353","23,2928175668263",
"24,8228104524092","26,3813154799118","27,9665482670056","29,576795159735
3","31,2104107828321","32,865815661106","34,5414939109064","36,2359910005
537","37,9479115785697","39,6759173684711","41,4187251288346","43,1751046
773006","44,9438769771451","46,7239122850217","48,5141283584553","50,3134
887216548","52,1210009882011","53,9357152391637","55,7567224552026","57,5
83153001211","59,4141751620684","61,2489937280806","63,0868486287004","64
,9270136131383","66,7687949764879","68,6115303300208","70,4545874143162",
"72,2973629539221","74,1392815522669","75,979794625567","77,8183793744999
","79,654537792444","81,487795709108","83,3177018684062","85,143827039461
","86,9657631596429","88,7831225085846","90,5955369121401","92,4026569752
783","94,2041513429391","95,9997059878972","97,7890235247138","99,5718225
488816","101,347837000291","103,116815550175","104,878521010719","106,632
729766543","108,379231227277","110,117827300508","111,84833188436","113,5
70570379017","115,284379216523","116,989605408185","118,68610610898","120

,373748198325","122,052407876644","123,721970277147","125,382329092283","
127,033386214324","128,675051389574","130,307241885707","131,929882171752
","133,542903610262","135,146244161227","136,739848097296","138,323665729
891","139,897653145818","141,461771953985","143,01598904185","144,5602763
41254","146,094610603266","147,61897318174","149,133349825232","150,63773
0476977","152,132109082638","153,616483405513","155,090854848946","156,55
5228285664","158,009611893773","159,454016999184","160,888457924208","162
,312951842105","163,727518637352","165,132180771427","166,526963153891","
167,911893018579","169,286999804701","170,65231504267","172,007872244484"
,"173,353706798483","174,68985586832","176,016358295988","177,33325450875
1","178,640586429825","179,938397392673","181,22673205878","182,505636338
764","183,775157316717"

Input 2

Jumlah Berkas	30
Point di Layar	30
Jarak celah-layar	100
Panjang Gelombang	532
Amplitudo	1

Coordinate

2
602
,-300,-299,-298,-297,-296,-295,-294,-293,-292,-291,-290,-289,-288,-287,-
286,-285,-284,-283,-282,-281,-280,-279,-278,-277,-276,-275,-274,-273,-
272,-271,-270,-269,-268,-267,-266,-265,-264,-263,-262,-261,-260,-259,-
258,-257,-256,-255,-254,-253,-252,-251,-250,-249,-248,-247,-246,-245,-
244,-243,-242,-241,-240,-239,-238,-237,-236,-235,-234,-233,-232,-231,-
230,-229,-228,-227,-226,-225,-224,-223,-222,-221,-220,-219,-218,-217,-
216,-215,-214,-213,-212,-211,-210,-209,-208,-207,-206,-205,-204,-203,-
202,-201,-200,-199,-198,-197,-196,-195,-194,-193,-192,-191,-190,-189,-
188,-187,-186,-185,-184,-183,-182,-181,-180,-179,-178,-177,-176,-175,-
174,-173,-172,-171,-170,-169,-168,-167,-166,-165,-164,-163,-162,-161,-
160,-159,-158,-157,-156,-155,-154,-153,-152,-151,-150,-149,-148,-147,-
146,-145,-144,-143,-142,-141,-140,-139,-138,-137,-136,-135,-134,-133,-
132,-131,-130,-129,-128,-127,-126,-125,-124,-123,-122,-121,-120,-119,-
118,-117,-116,-115,-114,-113,-112,-111,-110,-109,-108,-107,-106,-105,-
104,-103,-102,-101,-100,-99,-98,-97,-96,-95,-94,-93,-92,-91,-90,-89,-88,-
87,-86,-85,-84,-83,-82,-81,-80,-79,-78,-77,-76,-75,-74,-73,-72,-71,-70,-
69,-68,-67,-66,-65,-64,-63,-62,-61,-60,-59,-58,-57,-56,-55,-54,-53,-52,-
51,-50,-49,-48,-47,-46,-45,-44,-43,-42,-41,-40,-39,-38,-37,-36,-35,-34,-
33,-32,-31,-30,-29,-28,-27,-26,-25,-24,-23,-22,-21,-20,-19,-18,-17,-16,-
15,-14,-13,-12,-11,-10,-9,-8,-7,-6,-5,-4,-3,-2,-
1,0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,
27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,5
1,52,53,54,55,56,57,58,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75
,76,77,78,79,80,81,82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,97,98,99,
100,101,102,103,104,105,106,107,108,109,110,111,112,113,114,115,116,117,1
18,119,120,121,122,123,124,125,126,127,128,129,130,131,132,133,134,135,13
6,137,138,139,140,141,142,143,144,145,146,147,148,149,150,151,152,153,154
,155,156,157,158,159,160,161,162,163,164,165,166,167,168,169,170,171,172,
173,174,175,176,177,178,179,180,181,182,183,184,185,186,187,188,189,190,1
91,192,193,194,195,196,197,198,199,200,201,202,203,204,205,206,207,208,20
9,210,211,212,213,214,215,216,217,218,219,220,221,222,223,224,225,226,227
,228,229,230,231,232,233,234,235,236,237,238,239,240,241,242,243,244,245,
246,247,248,249,250,251,252,253,254,255,256,257,258,259,260,261,262,263,2

64,265,266,267,268,269,270,271,272,273,274,275,276,277,278,279,280,281,28
2,283,284,285,286,287,288,289,290,291,292,293,294,295,296,297,298,299,300

value

-
15,"736,654023143815","738,418335475005","740,197105193428","741,99045738
3968","743,798517603806","745,621411849735","747,45926652354","749,312208
395344","751,180364564834","753,063862420271","754,962829595184","756,877
393922656","758,807683387084","760,753826073303","762,715950112972","764,
694183628082","766,688654671475","768,699491164242","770,726820829856","7
72,770771124904","774,831469166273","776,909041654627","779,003614794 03",
"781,115314207531","783,244264848546","785,390590907856","787,55441571603
8","789,735861641113","791,935049981229","794,152100852155","796,38713306
9365","798,640264024476","800,911609555813","803,201283812839","805,50939
911419","807,83606579904","810,181392071537","812,545483837972","814,9284
44536415","817,330374958476","819,751373062866","822,191533780404","824,6
50948810131","827,129706406115","829,627891154609","832,145583741106","83
4,682860706904","837,239794194716","839,816451682883","842,412895707705",
"845,029183573398","847,665367049152","850,321492052765","852,99759832028
8","855,693719061094","858,409880597782","861,146101990269","863,90239464
3429","866,6787618976","869,475198601238","872,291690665002","875,1282145
96492","877,984737014857","880,861214144443","883,757591286626","886,6738
02268937","889,609768870566","892,565400223274","895,540592186719","898,5
35226697172","901,549171088538","904,582277384578","907,634381561172","91
0,705302777429","913,794842574402","916,902784040126","920,028890939634",
"923,172906808586","926,334554009074","929,513532746117","932,70952004333
","935,922168676166","939,151106061112","942,395933099123","945,656222971
572","948,931519886891","952,221337776047","955,525158934932","958,842432
611685","962,172573536909","965,514960394672","968,868934232147","972,233
796805653","975,608808860822","978,99318834456","982,38610854638","985,78
6696166662","989,19402930933","992,607135396352","996,024989001465","999,
446509600425","1002,87055923509","1006,29594008853","1009,72139196841","1
013,14558969576","1016,56714039628","1019,98458069133","1023,39637378563"
,"1026,80090644881","1030,19648588792","1033,58133650797","1036,953596557
68","1040,31131465759","1043,65244620779","1046,97484967261","1050,276282
73962","1053,55439835056","1056,80674060177","1060,03074051201","1063,223
71165576","1066,38284566011","1069,50520756392","1072,58773103801","1075,
62721346556","1078,62031088229","1081,56353277645","1084,45323674918","10
87,28562303625","1090,05672889295","1092,76242284456","1095,39839880565",
"1097,9601700722","1100,44306319179","1102,84221171789","1105,15254985562
","1107,36880600771","1109,4854962308","1111,4969176137","1113,3971415912
7","1115,18000720924","1116,83911435764","1118,36781699253","1119,7592163
6878","1121,00615430874","1122,10120653537","1123,0366761012","1123,80458
694836","1124,39667763878","1124,80439529788","1125,01888981989","1125,03
100838761","1124,8312903654","1124,4099626296","1123,75693540729","1122,8
6179870115","1121,71381938561","1120,3019390676","1118,61477281382","1116
,64060885607","1114,36740939579","1111,78281264038","1108,87413621498","1
105,62838210618","1102,03224330735","1098,07211234969","1093,73409191809"
,"1089,00400776767","1083,86742417371","1078,30966216652","1072,315820822
25","1065,87080190157","1058,95933815017","1051,56602559825","1043,675360
22096","1035,27177934737","1026,33970823315","1016,86361224014","1006,828
05509623","996,217763739741","985,017700284899","973,213141678391","960,7
8976765132","947,733757606467","934,031897116807","919,671694748085","904
,641509955412","888,930692841167","872,529736598587","855,430443502005","
837,626105340295","819,111699224147","799,88409972993","779,942308372239"
,"759,287701423192","737,924297118252","715,859043304867","693,1021266004
59","669,667304129187","645,572258901023","620,83897988073","595,49416776
6549","569,56966745733","543,102928130268","516,137491777605","488,723510
957209","460,918296396546","432,786894949784","404,402698240751","375,848
082127567","347,215076894917","318,606067814141","290,134525406363","261,
925764396682","234,117729954604","206,861809374502","180,323666856552","1

54,68409850067","130,139904020605","106,904771020277","85,2101669477701",
"65,3062330527754","47,4626738200527","31,969634434932","19,138557858145"
,"9,30301204882762","2,81947678011025","0,0680783467080812","1,4532592754
4934","7,40436892690647","18,3761596305824","34,8491717412573","57,329989
7565641","86,3513504149272","122,472082520932","166,276857147431","218,37
5725868892","279,403423820531","350,018413687865","430,901646249658","522
,755012865075","626,299465356934","742,272779143189","871,426936255734","
1014,52510610756","1172,33820357451","1345,64100619376","1535,20781509184
","1741,8076476828","1966,1989542562","2209,12385533491","2471,3019021431
7","2753,4233686956","3056,14209089237","3380,06787556775","3725,75851064
991","4093,71141639696","4484,35498700023","4898,03968159394","5335,02893
376046","5795,48995882663","6279,48454844217","6786,95995192397","7317,73
995342668","7871,51626292853","8447,84034705108","9046,11583260206","9665
,59162117119","10305,3558568491","10964,3308909145","11641,2693868872","1
2334,7517064519","13043,1847112014","13764,802106783","14497,6664447258",
"15239,6728829284","15988,5547884948","16741,8912463964","17497,116514460
6","18251,5314396681","19002,3168229967","19746,5486904631","20481,215397
0653","21203,2364585437","21909,4829738699","22596,7994697851","23262,026
9682367","23902,027048911","24513,7066529378","25094,0433509501","25640,1
107796641","26149,1039366046","26618,3640130475","27045,402441109","27427
,9238324804","27763,8474937662","28051,3272167681","28288,769061265","284
74,846872613","28608,5153064363","28689,0201672663","28715,9059065536","2
8689,0201672663","28608,5153064363","28474,846872613","28288,769061265","
28051,3272167681","27763,8474937662","27427,9238324804","27045,402441109"
,"26618,3640130475","26149,1039366046","25640,1107796641","25094,04335095
01","24513,7066529378","23902,027048911","23262,0269682367","22596,799469
7851","21909,4829738699","21203,2364585437","20481,2153970653","19746,548
6904631","19002,3168229967","18251,5314396681","17497,1165144606","16741,
8912463964","15988,5547884948","15239,6728829284","14497,6664447258","137
64,802106783","13043,1847112015","12334,7517064519","11641,2693868872","1
0964,3308909145","10305,3558568491","9665,59162117119","9046,11583260207"
,"8447,84034705108","7871,51626292853","7317,73995342668","6786,959951923
97","6279,48454844217","5795,48995882663","5335,02893376046","4898,039681
59394","4484,35498700023","4093,71141639696","3725,75851064991","3380,067
87556775","3056,14209089237","2753,4233686956","2471,30190214318","2209,1
2385533491","1966,1989542562","1741,8076476828","1535,20781509184","1345,
64100619376","1172,33820357452","1014,52510610756","871,426936255734","74
2,272779143189","626,299465356934","522,755012865075","430,901646249658",
"350,018413687865","279,403423820531","218,375725868892","166,27685714743
1","122,472082520932","86,3513504149272","57,3299897565641","34,849171741
2573","18,3761596305824","7,40436892690647","1,45325927544934","0,0680783
467080812","2,81947678011025","9,30301204882762","19,138557858145","31,96
9634434932","47,4626738200527","65,3062330527754","85,2101669477701","106
,904771020277","130,139904020605","154,68409850067","180,323666856552","2
06,861809374502","234,117729954605","261,925764396682","290,134525406363"
,"318,606067814141","347,215076894917","375,848082127567","404,4026982407
51","432,786894949784","460,918296396546","488,723510957209","516,1374917
77605","543,102928130268","569,56966745733","595,494167766549","620,83897
988073","645,572258901023","669,667304129187","693,102126600459","715,859
043304867","737,924297118252","759,287701423192","779,942308372239","799,
88409972993","819,111699224147","837,626105340295","855,430443502005","87
2,529736598587","888,930692841168","904,641509955412","919,671694748085",
"934,031897116808","947,733757606468","960,789767651321","973,21314167839
2","985,0177002849","996,217763739741","1006,82805509623","1016,863612240
14","1026,33970823315","1035,27177934737","1043,67536022096","1051,566025
59826","1058,95933815017","1065,87080190157","1072,31582082225","1078,309
66216652","1083,86742417371","1089,00400776767","1093,73409191809","1098,
07211234969","1102,03224330735","1105,62838210618","1108,87413621498","11
11,78281264038","1114,36740939579","1116,64060885607","1118,61477281382",
"1120,3019390676","1121,71381938561","1122,86179870115","1123,75693540729
","1124,4099626296","1124,8312903654","1125,03100838761","1125,0188898198
9","1124,80439529788","1124,39667763878","1123,80458694836","1123,0366761
012","1122,10120653537","1121,00615430874","1119,75921636878","1118,36781

699253","1116,83911435763","1115,18000720924","1113,39714159127","1111,49
69176137","1109,4854962308","1107,36880600771","1105,15254985562","1102,8
4221171789","1100,44306319179","1097,9601700722","1095,39839880565","1092
,76242284456","1090,05672889295","1087,28562303625","1084,45323674918","1
081,56353277644","1078,62031088229","1075,62721346556","1072,58773103801"
,"1069,50520756392","1066,38284566011","1063,22371165576","1060,030740512
01","1056,80674060177","1053,55439835056","1050,27628273962","1046,974849
67261","1043,65244620779","1040,31131465759","1036,95359655768","1033,581
33650797","1030,19648588792","1026,80090644881","1023,39637378563","1019,
98458069133","1016,56714039628","1013,14558969576","1009,72139196841","10
06,29594008853","1002,87055923509","999,446509600425","996,024989001465",
"992,607135396352","989,194029309331","985,786696166663","982,38610854638
","978,99318834456","975,608808860822","972,233796805653","968,8689342321
47","965,514960394672","962,172573536908","958,842432611685","955,5251589
34932","952,221337776047","948,931519886891","945,656222971572","942,3959
33099122","939,151106061111","935,922168676166","932,709520043329","929,5
13532746117","926,334554009073","923,172906808586","920,028890939634","91
6,902784040126","913,794842574402","910,705302777429","907,634381561173",
"904,582277384578","901,549171088537","898,535226697171","895,54059218671
9","892,565400223274","889,609768870566","886,673802268936","883,75759128
6625","880,861214144443","877,984737014857","875,128214596491","872,29169
0665001","869,475198601238","866,678761897599","863,902394643428","861,14
6101990268","858,409880597782","855,693719061095","852,997598320288","850
,321492052766","847,665367049153","845,029183573399","842,412895707706","
839,816451682883","837,239794194715","834,682860706903","832,145583741105
","829,627891154608","827,129706406114","824,65094881013","822,1915337804
04","819,751373062865","817,330374958476","814,928444536415","812,5454838
37972","810,181392071538","807,836065799041","805,50939911419","803,20128
3812839","800,911609555813","798,640264024475","796,387133069364","794,15
2100852154","791,935049981228","789,735861641112","787,554415716038","785
,390590907856","783,244264848545","781,115314207531","779,00361479403","7
76,909041654627","774,831469166273","772,770771124904","770,726820829856"
,"768,699491164242","766,688654671475","764,694183628081","762,7159501129
72","760,753826073303","758,807683387084","756,877393922656","754,9628295
95183","753,063862420271","751,180364564834","749,312208395344","747,4592
6652354","745,621411849735","743,798517603807","741,990457383969","740,19
7105193428","738,418335475005","736,654023143815"

Input 3

Jumlah Berkas	30
Point di Layar	30
Jarak celah-layar	100
Panjang Gelombang	656
Amplitudo	1

Coordinate

2
602
,-300,-299,-298,-297,-296,-295,-294,-293,-292,-291,-290,-289,-288,-287,-
286,-285,-284,-283,-282,-281,-280,-279,-278,-277,-276,-275,-274,-273,-
272,-271,-270,-269,-268,-267,-266,-265,-264,-263,-262,-261,-260,-259,-
258,-257,-256,-255,-254,-253,-252,-251,-250,-249,-248,-247,-246,-245,-
244,-243,-242,-241,-240,-239,-238,-237,-236,-235,-234,-233,-232,-231,-
230,-229,-228,-227,-226,-225,-224,-223,-222,-221,-220,-219,-218,-217,-
216,-215,-214,-213,-212,-211,-210,-209,-208,-207,-206,-205,-204,-203,-
202,-201,-200,-199,-198,-197,-196,-195,-194,-193,-192,-191,-190,-189,-

```
188,-187,-186,-185,-184,-183,-182,-181,-180,-179,-178,-177,-176,-175,-
174,-173,-172,-171,-170,-169,-168,-167,-166,-165,-164,-163,-162,-161,-
160,-159,-158,-157,-156,-155,-154,-153,-152,-151,-150,-149,-148,-147,-
146,-145,-144,-143,-142,-141,-140,-139,-138,-137,-136,-135,-134,-133,-
132,-131,-130,-129,-128,-127,-126,-125,-124,-123,-122,-121,-120,-119,-
118,-117,-116,-115,-114,-113,-112,-111,-110,-109,-108,-107,-106,-105,-
104,-103,-102,-101,-100,-99,-98,-97,-96,-95,-94,-93,-92,-91,-90,-89,-88,-
87,-86,-85,-84,-83,-82,-81,-80,-79,-78,-77,-76,-75,-74,-73,-72,-71,-70,-
69,-68,-67,-66,-65,-64,-63,-62,-61,-60,-59,-58,-57,-56,-55,-54,-53,-52,-
51,-50,-49,-48,-47,-46,-45,-44,-43,-42,-41,-40,-39,-38,-37,-36,-35,-34,-
33,-32,-31,-30,-29,-28,-27,-26,-25,-24,-23,-22,-21,-20,-19,-18,-17,-16,-
15,-14,-13,-12,-11,-10,-9,-8,-7,-6,-5,-4,-3,-2,-
1,0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,
27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,5
1,52,53,54,55,56,57,58,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75
,76,77,78,79,80,81,82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,97,98,99,
100,101,102,103,104,105,106,107,108,109,110,111,112,113,114,115,116,117,1
18,119,120,121,122,123,124,125,126,127,128,129,130,131,132,133,134,135,13
6,137,138,139,140,141,142,143,144,145,146,147,148,149,150,151,152,153,154
,155,156,157,158,159,160,161,162,163,164,165,166,167,168,169,170,171,172,
173,174,175,176,177,178,179,180,181,182,183,184,185,186,187,188,189,190,1
91,192,193,194,195,196,197,198,199,200,201,202,203,204,205,206,207,208,20
9,210,211,212,213,214,215,216,217,218,219,220,221,222,223,224,225,226,227
,228,229,230,231,232,233,234,235,236,237,238,239,240,241,242,243,244,245,
246,247,248,249,250,251,252,253,254,255,256,257,258,259,260,261,262,263,2
64,265,266,267,268,269,270,271,272,273,274,275,276,277,278,279,280,281,28
2,283,284,285,286,287,288,289,290,291,292,293,294,295,296,297,298,299,300

Value

-
15,"1084,69451683078","1084,0198979677","1083,33390532887","1082,63632577
212","1081,92694174236","1081,20553117303","1080,47186738538","1079,72571
898527","1078,96684975772","1078,19501855896","1077,40997920597","1076,61
14803635","1075,79926542848","1074,97307241178","1074,13263381725","1073,
27767651795","1072,4079216296","1071,52308438109","1070,62287398206","106
9,70699348749","1068,77513965913","1067,82700282386","1066,86226672887","
1065,88060839344","1064,88169795757","1063,86519852706","1062,83076601523
","1061,77804898104","1060,70668846367","1059,61631781344","1058,50656251
896","1057,37704003055","1056,22735957976","1055,05712199503","1053,86591
951325","1052,65333558744","1051,41894469018","1050,16231211295","1048,88
299376116","1047,580535945","1046,25447516585","1044,9043378983","1043,52
964036774","1042,12988832345","1040,70457680706","1039,2531899165","1037,
77520056524","1036,2700702369","1034,73724873506","1033,17617392845","103
1,58627149123","1029,96695463862","1028,31762385764","1026,63766663307","
1024,92645716863","1023,18335610338","1021,40771022328","1019,5988521681"
,"1017,75610013362","1015,87875756908","1013,96611287024","1012,017439067
74","1010,03199351116","1008,00901754879","1005,94773620315","1003,847357
84256","1001,70707384876","999,526058280938","997,303467536213","995,0384
40006927","992,730095734941","990,377536063247","987,979843285222","985,5
36080291863","983,045290217411","980,506496083785","977,918700444285","97
5,280885027107","972,592010379206","969,851015511135","967,056817543528",
"964,208311355948","961,304369238897","958,343840549845","955,32555137420
4","952,248304192249","949,110877553102","945,912025756919","942,65047854
6575","939,324940810222","935,934092296189","932,476587341826","928,95105
4618016","925,3560968912","921,690290804897","917,952186682891","914,1403
08356356","910,253153017401","906,289191101687","902,246866202996","898,1
24595022748","893,920767357804","889,633746130038","885,261867461454","88
0,80344079887","876,25674909252","871,620049033158","866,891571352647","8
62,069521193312","857,152078551724","852,137398802975","847,023613311905"
,"841,808830138214","836,49113484284","831,068591403488","825,53924324776
","819,901114412857","814,152210841475","808,290521824125","802,314021598
818","796,220671119757","790,008420007475","783,675208693654","777,218970
```

774741","770,637635589418","763,929131035906","757,09138664622","750,1223
36935472","743,019925045606","735,78210670411","728,406854519586","720,89
2162637472","713,236051780641","705,436574701197","697,491822071423","689
,399928843568","681,159081110031","672,767523497419","664,22356713003","6
55,525598200487","646,672087187511","637,661598763269","628,492802435268"
,"619,164483970417","609,675557651741","600,02507942116","590,21226096488
1","580,236484801201","570,097320433971","559,794541638539","549,32814495
0767","538,69836943365","527,905717800161","516,950978975254","505,835252
184392","494,559972660657","483,126939067276","471,538342737451","459,796
798838537","447,905379572972","435,867649533895","423,687703339067","411,
370205672525","398,920433869394","386,344323185329","373,648514898259","3
60,84040739638","347,928210412601","334,921002572021","321,828792425271",
"308,662583146829","295,434441083479","282,157568344087","268,84637962752
4","255,516583490995","242,185268265999","228,870992833668","215,59388247
5158","202,375730015928","189,240102485134","176,212453512699","163,32024
1686878","150,59305509396","138,062742259165","125,763549703384","113,732
266324032","102,008374799733","90,6342102073787","79,655126026209","69,11
96676864271","59,0797537992704","49,5908651809264","40,7122417538639","32
,507087375552","25,0427826057379","18,3911053788917","12,6284594976399","
7,83611080538205","4,10043083127958","1,51314762778606","0,17160343925084
","0,179018749210208","1,6447621531426l","4,68462539203245","9,4211027594
0992","15,9836739599544","24,5090893506386","35,14165633515","48,03352550
83996","63,3449749598279","81,2446909415238","101,91004288958","125,52735
0554419","152,292140748006","182,409390953044","216,093756761757","253,56
9779820326","295,072072650261","340,845476401182","391,145187262157","446
,236846922861","506,396592133697","571,911058068609","643,077329848953","
720,202836245461","803,60517924271","893,611892831769","990,560124097906"
,"1094,79622939802","1206,67527818443","1326,56045683567","1454,822364710
75","1591,83819455949","1737,99078941007","1893,66756812521","2059,259311
98389","2235,15880491698","2421,75932041612","2619,45294865801","2828,628
75805357","3049,67078625642","3282,95585665854","3528,85121757569","3787,
71200268947","4059,87851287631","4345,67332132276","4645,39820580396","49
59,33091419109","5287,72177164966","5630,79014058947","5988,72074721618",
"6361,65989150029","6749,71156049943","7152,93346821832","7571,3330485349
7","8004,8634311221","8453,41943370456","8916,83360736433","9394,87237487
688","9887,23230517139","10393,5365698844","10913,3316305472","11446,0842
071328","11991,1785804137","12547,9142817564","13115,5042245362","13693,0
733312036","14279,6577091128","14874,204426458","15475,571937001","16082,
5311986735","16693,7675265608","17307,8832152112","17923,4009586758","185
38,768089175","19152,3616468812","19762,4942840481","20367,4209967128","2
0965,3466665605","21554,4343844114","22132,8145153247","22698,5944537017"
,"23249,869005202","23784,7313209729","24301,2842988602","24797,652356132
9","25271,993469049","25722,5113665255","26147,46775846","26545,194474080
8","26914,1053822266","27252,7079638287","27559,6144071806","27833,552098
906","28073,3733878922","28278,0645058357","28446,7535363831","28578,7173
35037","28673,3873138899","28730,3540186658","28749,3704402528","28730,35
40186658","28673,3873138899","28578,717335037","28446,7535363831","28278,
0645058357","28073,3733878922","27833,552098906","27559,6144071806","2725
2,7079638287","26914,1053822266","26545,1944740808","26147,4677584599","2
5722,5113665255","25271,993469049","24797,6523561328","24301,2842988602",
"23784,7313209729","23249,869005202","22698,5944537017","22132,8145153247
","21554,4343844114","20965,3466665605","20367,4209967128","19762,4942840
481","19152,3616468812","18538,768089175","17923,4009586758","17307,88321
52112","16693,7675265608","16082,5311986735","15475,571937001","14874,204
426458","14279,6577091128","13693,0733312036","13115,5042245362","12547,9
142817564","11991,1785804137","11446,0842071328","10913,3316305472","1039
3,5365698844","9887,23230517139","9394,87237487688","8916,83360736433","8
453,41943370455","8004,8634311221","7571,33304853497","7152,93346821832",
"6749,71156049943","6361,65989150029","5988,72074721618","5630,7901405894
7","5287,72177164966","4959,33091419109","4645,39820580396","4345,6733213
2276","4059,87851287631","3787,71200268947","3528,85121757569","3282,9558
5665854","3049,67078625642","2828,62875805358","2619,45294865801","2421,7
5932041612","2235,15880491698","2059,25931198389","1893,66756812521","173

7,99078941007","1591,83819455949","1454,82236471075","1326,56045683567","1206,67527818443","1094,79622939802","990,560124097905","893,611892831769","803,60517924271","720,202836245461","643,077329848953","571,91105806861","506,396592133697","446,236846922861","391,145187262157","340,845476401182","295,072072650261","253,569779820326","216,093756761757","182,409390953044","152,292140748006","125,527350554419","101,91004288958","81,2446909415238","63,3449749598279","48,0335255083996","35,14165633515","24,5090893506386","15,9836739599544","9,42110275940993","4,68462539203245","1,64476215314261","0,179018749210208","0,17160343925084","1,513147627786","4,10043083127958","7,83611080538205","12,6284594976399","18,3911053788917","25,0427826057379","32,5070873755552","40,7122417538639","49,5908651809265","59,0797537992704","69,1196676864271","79,655126026209","90,6342102073788","102,008374799733","113,732266324033","125,763549703384","138,062742259166","150,59305509396","163,320241686878","176,212453512699","189,240102485134","202,375730015928","215,593882475158","228,870992833668","242,185268265999","255,516583490995","268,846379627524","282,157568344087","295,434441083479","308,662583146829","321,828792425271","334,921002572021","347,928210412602","360,840407396381","373,648514898259","386,344323185329","398,920433869394","411,370205672525","423,687703339067","435,867649533895","447,905379572972","459,796798838537","471,538342737451","483,126939067276","494,559972660657","505,835252184392","516,950978975253","527,905717800161","538,69836943365","549,328144950767","559,794541638539","570,097320433971","580,236484801201","590,212260964881","600,02507942116","609,675557651741","619,164483970417","628,492802435268","637,661598763269","646,67208718751","655,525598200487","664,223567130029","672,767523497418","681,15908111003","689,399928843568","697,491822071422","705,436574701197","713,236051780641","720,892162637472","728,406854519586","735,78210670411","743,019925045606","750,122336935472","757,09138664622","763,929131035907","770,637635589418","777,218970774742","783,675208693654","790,008420007475","796,220671119756","802,314021598817","808,290521824125","814,152210841474","819,901114412857","825,53924324776","831,068591403488","836,49113484284","841,808830138214","847,023613311905","852,137398802975","857,152078551724","862,069521193312","866,891571352647","871,620049033158","876,25674909252","880,80344079887","885,261867461454","889,633746130039","893,920767357804","898,124595022748","902,246866202997","906,289191101688","910,253153017401","914,140308356356","917,952186682891","921,690290804897","925,3560968912","928,951054618016","932,476587341826","935,934092296189","939,324940810223","942,650478546575","945,912025756919","949,110877553102","952,248304192249","955,325551374204","958,343840549845","961,304369238897","964,208311355948","967,056817543528","969,851015511135","972,592010379205","975,280885027106","977,918700444285","980,506496083785","983,04529021741","985,536080291862","987,979843285221","990,377536063246","992,73009573494","995,038440006926","997,303467536212","999,526058280938","1001,70707384876","1003,84735784256","1005,94773620315","1008,00901754879","1010,03199351116","1012,01743906774","1013,96611287024","1015,87875756908","1017,75610013362","1019,5988521681","1021,40771022328","1023,18335610338","1024,92645716863","1026,63766663307","1028,31762385764","1029,96695463862","1031,58627149123","1033,17617392845","1034,73724873506","1036,2700702369","1037,77520056524","1039,2531899165","1040,70457680706","1042,12988832345","1043,52964036774","1044,9043378983","1046,25447516585","1047,580535945","1048,88299376116","1050,16231211295","1051,41894469018","1052,65333558744","1053,86591951325","1055,05712199503","1056,22735957976","1057,37704003055","1058,50656251896","1059,61631781344","1060,70668846367","1061,77804898104","1062,83076601523","1063,86519852706","1064,88169795757","1065,88060839344","1066,86226672887","1067,82700282386","1068,77513965913","1069,70699348749","1070,62287398206","1071,52308438109","1072,4079216296","1073,27767651795","1074,13263381725","1074,973072411178","1075,79926542848","1076,6114803635","1077,40997920597","1078,19501855896","1078,96684975772","1079,72571898526","1080,47186738538","1081,20553117303","1081,92694174236","1082,63632577212","1083,33390532887","1084,0198979677","1084,69451683078"

Lampiran 4 : Kode Program Difraksi Zaenuri A, 2008

```
unit uTesHitung;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
Forms,
  Dialogs, Math, SpectraLibrary, ComCtrls, StdCtrls, ExtCtrls,
  TeEngine, Series, TeeProcs, Chart,
  Grids;

 type
   TRGBColor = record
     Red,
     Green,
     Blue : Byte;
   end;

   THSBColor = record
     Hue,
     Saturnation,
     Brightness : Double;
   end;

type
  TfrmSImulHitung = class(TForm)
    PageControl1: TPageControl;
    tabSingleSlit: TTabSheet;
    tabDoubleSlit: TTabSheet;
    UpDown1: TUpDown;
    edPanjangGelombang: TLabeledEdit;
    edLebarCelah: TLabeledEdit;
    edJarakKeLayar: TLabeledEdit;
    UpDown3: TUpDown;
    edIMax: TLabeledEdit;
    TrackBar1: TTrackBar;
    UpDown4: TUpDown;
    btnHitung: TButton;
    lblAlfa: TLabel;
    lblITheta: TLabel;
    edTheta: TLabeledEdit;
    UpDown5: TUpDown;
    shpColor: TShape;
    btnSimulasi: TButton;
    Panel1: TPanel;
    chartSingle: TChart;
    Series1: TFastLineSeries;
    edLambdaDouble: TLabeledEdit;
    edLEbarCelahDouble: TLabeledEdit;
    edSudutTesDouble: TLabeledEdit;
```

```
    edIMaxDouble: TLabeledEdit;
    edJarakKeLayarDouble: TLabeledEdit;
    UpDown6: TUpDown;
    UpDown7: TUpDown;
    UpDown9: TUpDown;
    UpDown10: TUpDown;
    shpColorDouble: TShape;
    TrackBar2: TTrackBar;
    lblAlfaDouble: TLabel;
    lblThetaDouble: TLabel;
    Panel2: TPanel;
    chartDouble: TChart;
    btnDoubleSimul: TButton;
    Series2: TFastLineSeries;
    edJarakCelah: TLabeledEdit;
    edSudutMin: TLabeledEdit;
    edSudutMax: TLabeledEdit;
    UpDown11: TUpDown;
    UpDown12: TUpDown;
    edSudutMinDouble: TLabeledEdit;
    UpDown13: TUpDown;
    UpDown14: TUpDown;
    edSudutMaxDouble: TLabeledEdit;
    btnGrafik1: TButton;
    btnGrafik2: TButton;
    btnHitungDouble: TButton;
    imgDouble: TImage;
    imgSingle: TImage;
    Image1: TImage;
    sgMinMaxSingle: TStringGrid;
    sgMinMaxDouble: TStringGrid;
    StatusBar1: TStatusBar;
    Label1: TLabel;
    Label2: TLabel;
    TabSheet1: TTabSheet;
    StringGrid1: TStringGrid;
    Edit1: TEdit;
    procedure btnHitungClick(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure edPanjangGelombangChange(Sender: TObject);
    procedure edIMaxChange(Sender: TObject);
    procedure btnSimulasiClick(Sender: TObject);
    procedure edLambdaDoubleChange(Sender: TObject);
    procedure btnDoubleSimulClick(Sender: TObject);
    procedure btnGrafik1Click(Sender: TObject);
    procedure btnGrafik2Click(Sender: TObject);
    procedure btnHitungDoubleClick(Sender: TObject);
    procedure TrackBar1Change(Sender: TObject);
    procedure TrackBar2Change(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmSImulHitung: TfrmSImulHitung;

implementation

{$R *.dfm}

//untuk mengetes perhitungan
```

```
procedure TfrmSImulHitung.btnHitungClick(Sender: TObject);
var
  a, lambda, alfa, theta, itheta, imax: extended;
begin
  a:=StrToInt(edLebarCelah.Text)*power(10,-9); //lebar celah dlm m
  lambda:=StrToInt(edPanjangGelombang.Text)*power(10,-9); //panjang
gelombang
  //dlm m
  if (a<lambda) then
  begin
    MessageDlg('Slit width must equal or more than wavelength!', mtError,
[
      mbOK], 0);
    exit;
  end;

  theta:=DegToRad(StrToInt(edTheta.Text)); //sudut dlm radian
  imax:=StrToInt(edIMax.Text); //intensitas maksimum

  alfa:=((Pi*a)/lambda)*sin(theta); //hitung alfa
  lblAlfa.Caption:='Alfa = '+FloatToStr(alfa); //tampilkan pada label

  if (alfa=0) then //jika alfa=0
  begin
    itheta:=imax;
  end else        //jika alfa tidak sama dengan nol
  begin
    itheta:=imax*sqr(sin(alfa)/alfa);
  end;
  lblITheta.Caption:='I theta = '+FloatToStr(itheta); //tampilkan Itheta
pada
  //label
end;

//dijalankan ketika aplikasi mulai
procedure TfrmSImulHitung.FormCreate(Sender: TObject);
var
  R, G, B:Byte;
begin
  //ubtuk menampilkan kotak warna pada single slit
  WavelengthToRGB(strToInt(edPanjangGelombang.Text), R, G, B);
  shpColor.Brush.Color:=StringToColor('$00'+IntToHex(B, 2)+IntToHex(G,
2)+
    IntToHex(R, 2));
  //ubtuk menampilkan kotak warna pada double slit
  WavelengthToRGB(strToInt(edLambdaDouble.Text), R, G, B);
  shpColorDouble.Brush.Color:=StringToColor('$00'+IntToHex(B,
2)+IntToHex(G, 2)+
    IntToHex(R, 2));
end;

//mengupdate kotak warna
procedure TfrmSImulHitung.edPanjangGelombangChange(Sender: TObject);
var
  R, G, B:Byte;
begin
  if (Length(edPanjangGelombang.Text)<=0) then
    exit;
  if ((strToInt(edPanjangGelombang.Text)<380) or (strToInt(
    edPanjangGelombang.Text)>750)) then
    exit;
  WavelengthToRGB(strToInt(edPanjangGelombang.Text), R, G, B);
```

```
  shpColor.Brush.Color:=StringToColor('$00'+IntToHex(B, 2)+IntToHex(G,
2)+
    IntToHex(R, 2));

  //Series1.Color:=shpColor.Brush.Color;
  TrackBar1.Position:=StrToInt(edPanjangGelombang.Text);
end;

//mengupdate sumbu Y (intensitas relatif)
procedure TfrmSImulHitung.edIMaxChange(Sender: TObject);
begin
  //chartSingle.Axes.Left.Minimum:=0;
  //chartSingle.Axes.Left.Maximum:=StrToInt(edIMax.Text);
  chartSingle.Invalidate;
end;

//untuk menggambar grafik pada single slit
procedure TfrmSImulHitung.btnSimulasiClick(Sender: TObject);
var
  a, lambda, alfa, theta, itheta, imax, mLambda, mTheta, y: extended;
  i, lebar: integer;
  jarakKeLayar: double;
  BmpSpectrum: TBitmap;
  r, g, b:byte;
  Z: Array[-1000..1000,-100..100] of Extended;
begin
  a:=StrToInt(edLebarCelah.Text)*power(10,-9); //lebar celah dlm m
  lambda:=StrToInt(edPanjangGelombang.Text)*power(10,-9); //panjang
gelombang
  //dlm m
  if (a<lambda) then
  begin
    MessageDlg('Slit width must equal or more than wavelength!', mtError,
[
      mbOK], 0);
      exit;
  end;
  //disable tombol simulasi dulu
  btnSimulasi.Enabled:=false;
  edSudutMin.Enabled:=false;
  edSudutMax.Enabled:=false;

  Series1.Clear;   //bersihkan grafik

  //untuk gambar spektrum
  BmpSpectrum:=TBitmap.Create;
  BmpSpectrum.Height:=imgSingle.Height;
  BmpSpectrum.Width:=(StrToInt(edSudutMax.Text)*25)-
(StrToInt(edSudutMin.Text)*
    25);
  BmpSpectrum.PixelFormat := pf24bit;
  BmpSpectrum.Canvas.Pen.Style:=psSolid;

  imax:=StrToInt(edIMax.Text); //intensitas maksimum

  lebar:=0; //untuk iterasi lebar spektrum
  //iterasi dari sudut min ke sudut max
  for i := (StrToInt(edSudutMin.Text)*25) to
(StrToInt(edSudutMax.Text)*25) do
  begin
    application.ProcessMessages;        //biar gak kelihatan hang
    theta:=DegToRad(i/25); //theta
    alfa:=((Pi*a)/lambda)*sin(theta); //hitung alfa
```

```
    lblAlfa.Caption:='Alfa = '+FloatToStr(alfa);

    if (alfa=0) then   //jika alfa=0
    begin
      itheta:=imax;
    end else       //jika alfa tidak sama dengan nol
    begin
      itheta:=imax*sqr(sin(alfa)/alfa);
    end;
    lblITheta.Caption:='I theta = '+FloatToStr(itheta); //ini yang jadi
alamat
     Edit1.Text:=IntToStr(i);
     StringGrid1.Cells[1,i+2250]:= FloatToStr(i/25);
     StringGrid1.Cells[2,i+2250]:= FloatToStr(itheta);
     StringGrid1.Cells[0,i+2250]:= FloatToStr(theta);
     //Z[1,i]:= itheta;

    Series1.AddXY(i/25, itheta);     //gambar titiknya di grafik

    //untuk menggambar spektrum pola difraksi
    BmpSpectrum.Canvas.MoveTo(lebar,0);
    WavelengthToRGB(lambda*power(10,9),r,g,b);
    BmpSpectrum.Canvas.Pen.Color:=RGB(round(r*itheta), round(g*itheta),
round(b*itheta));
    BmpSpectrum.Canvas.LineTo(lebar, BmpSpectrum.Height-1);

    lebar:=lebar+1;  //naikkan nilai lebar spektrum, untuk iterasi
  end;
  imgSingle.Picture.Graphic:=BmpSpectrum; //tampilkan gambar spektrumnya

  //bikin single slit minima
  jarakKeLayar:=StrToFloat(edJarakKeLayar.Text)*power(10, -2); //jarak ke
layar
  //dlm meter
  sgMinMaxSingle.RowCount:=1;
  sgMinMaxSingle.ColCount:=3;
  sgMinMaxSingle.Cells[0,0]:='Minima';
  sgMinMaxSingle.Cells[1,0]:='Theta(degree)';
  sgMinMaxSingle.Cells[2,0]:='Displacement(cm)';
  mLambda:=1*lambda;
  i:=1;
  mTheta:=radtodeg(ArcSin(mLambda/a)); //dalam derajat
  y:=jarakKeLayar*tan(DegToRad(mTheta)); //displacement

  //ulang sampai melebihi batas sudut
  while (mTheta<(StrToInt(edSudutMax.Text))) do
  begin
    //tampilkan di tabel
    sgMinMaxSingle.RowCount:=sgMinMaxSingle.RowCount+1;
    sgMinMaxSingle.FixedRows:=1;
    sgMinMaxSingle.Cells[0,i]:=IntToStr(i);
    sgMinMaxSingle.Cells[1,i]:=FloatToStr(mTheta);
    sgMinMaxSingle.Cells[2,i]:=FloatToStr(y*100);
    inc(i);
    //hitung ulang
    mLambda:=i*lambda;
    if (abs(mLambda/a)<1) then
    begin
      mTheta:=RadToDeg(ArcSin(mLambda/a)); //dalam derajat
      y:=jarakKeLayar*tan(DegToRad(mTheta)); //displacement
    end else
    begin
      break;
```

```
    end;
  end;

  //enable tombol simulasi
  btnSimulasi.Enabled:=true;
  edSudutMin.Enabled:=true;
  edSudutMax.Enabled:=true;
end;

//untuk mengupdate kotak warna pada double slit
procedure TfrmSImulHitung.edLambdaDoubleChange(Sender: TObject);
var
  R, G, B:Byte;
begin
  if (Length(edLambdaDouble.Text)<=0) then
    exit;
  if ((strToInt(edLambdaDouble.Text)<380) or (strToInt(
    edLambdaDouble.Text)>750)) then
    exit;
  WavelengthToRGB(strToInt(edLambdaDouble.Text), R, G, B);
  shpColorDouble.Brush.Color:=StringToColor('$00'+IntToHex(B,
2)+IntToHex(G, 2)+
    IntToHex(R, 2));

  //Series2.Color:=shpColorDouble.Brush.Color;
  TrackBar2.Position:=StrToInt(edLambdaDouble.Text);
end;

//untuk menggambar grafik pada double slit
procedure TfrmSImulHitung.btnDoubleSimulClick(Sender: TObject);
var
  a, d, lambda, alfa, beta, theta, itheta, imax, mLambda, ThetaMax,
ThetaMin, yMax, yMin: extended;
  i, lebar: integer;
  jarakKeLayar: double;
  BmpSpectrum: TBitmap;
  r, g, b:byte;
begin
  a:=StrToInt(edLEbarCelahDouble.Text)*power(10,-9); //lebar celah dlm m
  lambda:=StrToInt(edLambdaDouble.Text)*power(10,-9); //panjang gelombang
dlm m
  if (a<lambda) then
  begin
    MessageDlg('Slit width must equal or more than wavelength!', mtError,
[
      mbOK], 0);
    exit;
  end;
  //disable tombol simulasi dulu
  btnDoubleSimul.Enabled:=false;
  edSudutMinDouble.Enabled:=false;
  edSudutMaxDouble.Enabled:=false;

  Series2.Clear; //bersihkan grafik

  //untuk gambar spektrum
  BmpSpectrum:=TBitmap.Create;
  BmpSpectrum.Height:=imgDouble.Height;
  BmpSpectrum.Width:=(StrToInt(edSudutMaxDouble.Text)*25)-
(StrToInt(edSudutMinDouble.Text)*
    25);
  BmpSpectrum.PixelFormat := pf24bit;
  BmpSpectrum.Canvas.Pen.Style:=psSolid;
```

```
  d:=StrToInt(edJarakCelah.Text)*power(10,-9); //jarak antar celah dlm m
  imax:=StrToInt(edIMaxDouble.Text);             //intensitas maksimum

  lebar:=0;//untuk iterasi lebar spektrum
  //iterasi dari sudut min ke sudut max
  for i := StrToInt(edSudutMinDouble.Text)*25 to StrToInt(
    edSudutMaxDouble.Text)*25 do
  begin
    application.ProcessMessages; //biar gak kelihatan hang

    theta:=DegToRad(i/25);   //sudutnya
    alfa:=((Pi*a)/lambda)*sin(theta); //hitung alfa
    beta:=((Pi*d)/lambda)*sin(theta); //hitung beta
    lblAlfaDouble.Caption:='Alfa = '+FloatToStr(alfa);

    if (alfa=0) then       //jika alfa=0
    begin
      itheta:=imax;
    end else               //jika alfa tidak sama dengan nol
    begin
      itheta:=imax*sqr(sin(alfa)/alfa)*sqr(cos(beta));
      //itheta:=imax*(power((sin(alfa)/alfa),2))*(power(cos(beta),2));
    end;
    lblThetaDouble.Caption:='I theta = '+FloatToStr(itheta);

    Series2.AddXY(i/25, itheta);      //gambar titiknya di grafik

    //untuk menggambar spektrum pola difraksi
    BmpSpectrum.Canvas.MoveTo(lebar,0);
    WavelengthToRGB(lambda*power(10,9),r,g,b);
    BmpSpectrum.Canvas.Pen.Color:=RGB(round(r*itheta), round(g*itheta),
round(b*itheta));
    BmpSpectrum.Canvas.LineTo(lebar, BmpSpectrum.Height-1);

    lebar:=lebar+1;  //naikkan nilai lebar spektrum, untuk iterasi
  end;
  imgDouble.Picture.Graphic:=BmpSpectrum;    //tampilkan gambar
spektrumnya

  //bikin double slit minima/maxima
  jarakKeLayar:=StrToFloat(edJarakKeLayarDouble.Text)*power(10, -2);
//jarak ke layar
  //dlm meter
  sgMinMaxDouble.RowCount:=1;
  sgMinMaxDouble.ColCount:=5;
  sgMinMaxDouble.Cells[0,0]:='Minima/Maxima';
  sgMinMaxDouble.Cells[1,0]:='ThetaMax';
  sgMinMaxDouble.Cells[2,0]:='ThetaMin';
  sgMinMaxDouble.Cells[3,0]:='YMax';
  sgMinMaxDouble.Cells[4,0]:='YMin';
  mLambda:=0*lambda;
  i:=0;
  ThetaMax:=radtodeg(ArcSin(mLambda/d)); //dalam derajat
  yMax:=jarakKeLayar*tan(DegToRad(ThetaMax)); //displacement
  try
    ThetaMin:=radtodeg(ArcSin(((i+0.5)*lambda)/d)); //dalam derajat
  except
    MessageDlg('Wrong distance between slit input!', mtError, [mbOK],0);
    //enable tombol simulasi
    btnDoubleSimul.Enabled:=true;
    edSudutMinDouble.Enabled:=true;
    edSudutMaxDouble.Enabled:=true;
```

```
    exit;
  end;

  yMin:=jarakKeLayar*tan(DegToRad(ThetaMin)); //displacement

  //ulang sampai melebihi batas sudut
  while (ThetaMin<(StrToInt(edSudutMaxDouble.Text))) do
  begin
    //tampilkan di tabel
    sgMinMaxDouble.RowCount:=sgMinMaxDouble.RowCount+1;
    sgMinMaxDouble.FixedRows:=1;
    sgMinMaxDouble.Cells[0,i+1]:=IntToStr(i);
    sgMinMaxDouble.Cells[1,i+1]:=FloatToStr(ThetaMax);
    sgMinMaxDouble.Cells[2,i+1]:=FloatToStr(ThetaMin);
    sgMinMaxDouble.Cells[3,i+1]:=FloatToStr(YMax*100);
    sgMinMaxDouble.Cells[4,i+1]:=FloatToStr(YMin*100);
    inc(i);
    //hitung ulang
    mLambda:=i*lambda;
    if (abs(((i+0.5)*lambda)/d)<1) then
    begin
      ThetaMax:=radtodeg(ArcSin(mLambda/d)); //dalam derajat
      yMax:=jarakKeLayar*tan(DegToRad(ThetaMax)); //displacement
      ThetaMin:=radtodeg(ArcSin(((i+0.5)*lambda)/d)); //dalam derajat
      yMin:=jarakKeLayar*tan(DegToRad(ThetaMin)); //displacement
    end else
    begin
      break;
    end;
  end;

  //enable tombol simulasi
  btnDoubleSimul.Enabled:=true;
  edSudutMinDouble.Enabled:=true;
  edSudutMaxDouble.Enabled:=true;
end;

//untuk menggambar grafik sin(alfa)/alfa kuadrat
procedure TfrmSImulHitung.btnGrafik1Click(Sender: TObject);
var
  a, lambda, alfa, theta, itheta, imax: extended;
  i: integer;
begin
  Series2.Clear;

  a:=StrToInt(edLEbarCelahDouble.Text)*power(10,-9); //dlm m
  lambda:=StrToInt(edLambdaDouble.Text)*power(10,-9); //dlm m
  if (a<lambda) then
  begin
    MessageDlg('Slit width must equal or more than wavelength!', mtError,
[
      mbOK], 0);
    exit;
  end;

  imax:=StrToInt(edIMaxDouble.Text);
  for i := StrToInt(edSudutMinDouble.Text)*25 to StrToInt(
    edSudutMaxDouble.Text)*25 do
  begin
    theta:=DegToRad(i/25);
    alfa:=((Pi*a)/lambda)*sin(theta); //hitung alfa
    lblAlfaDouble.Caption:='Alfa = '+FloatToStr(alfa);
```

```
    if (alfa=0) then
    begin
      itheta:=imax;
    end else
    begin
      itheta:=imax*(sqr(sin(alfa)/alfa));
    end;
    lblThetaDouble.Caption:='I theta = '+FloatToStr(itheta);

    Series2.AddXY(i/25, itheta);
  end;
end;

//untuk menggambar cos kuadrat beta
procedure TfrmSImulHitung.btnGrafik2Click(Sender: TObject);
var
  d, lambda, beta, theta, itheta, imax: extended;
  i: integer;
begin
  Series2.Clear;

  d:=StrToInt(edJarakCelah.Text)*power(10,-9); //dlm m
  lambda:=StrToInt(edLambdaDouble.Text)*power(10,-9); //dlm m
  imax:=StrToInt(edIMaxDouble.Text);
  for i := StrToInt(edSudutMinDouble.Text)*25 to StrToInt(
    edSudutMaxDouble.Text)*25 do
  begin
    theta:=DegToRad(i/25);
    beta:=((Pi*d)/(lambda))*sin(theta); //hitung beta
    lblAlfaDouble.Caption:='Beta = '+FloatToStr(beta);

    if (beta=0) then
    begin
      itheta:=imax;
    end else
    begin
      itheta:=imax*sqr(cos(beta));
    end;
    lblThetaDouble.Caption:='I theta = '+FloatToStr(itheta);

    Series2.AddXY(i/25, itheta);
  end;
end;

//untuk tes perhitungan
procedure TfrmSImulHitung.btnHitungDoubleClick(Sender: TObject);
var
  a, d, lambda, alfa, beta, theta, itheta, imax: extended;
  i: integer;
begin
  a:=StrToInt(edLEbarCelahDouble.Text)*power(10,-9); //dlm m
  d:=StrToInt(edJarakCelah.Text)*power(10,-9); //dlm m
  lambda:=StrToInt(edLambdaDouble.Text)*power(10,-9); //dlm m
  if (a<lambda) then
  begin
    MessageDlg('Slit width must equal or more than wavelength!', mtError,
[
      mbOK], 0);
    exit;
  end;

  imax:=StrToInt(edIMaxDouble.Text);
  i:=StrToInt(edSudutTesDouble.Text);
```

```
  theta:=DegToRad(i/25);
  alfa:=((Pi*a)/lambda)*sin(theta); //hitung alfa
  beta:=((Pi*d)/lambda)*sin(theta); //hitung beta
  lblAlfaDouble.Caption:='Alfa = '+FloatToStr(alfa)+#13+'Beta =
'+FloatToStr(beta);

  if (alfa=0) then
  begin
    itheta:=imax;
  end else
  begin
    itheta:=imax*sqr(sin(alfa)/alfa)*sqr(cos(beta));
  end;
  lblThetaDouble.Caption:='I theta = '+FloatToStr(itheta);
end;

//update nilai panjang gelombang pada single slit berdasar perubahan
trackbar
procedure TfrmSImulHitung.TrackBar1Change(Sender: TObject);
begin
  edPanjangGelombang.Text:=IntToStr(TrackBar1.Position);
end;

//update nilai panjang gelombang pada double slit berdasar perubahan
trackbar
procedure TfrmSImulHitung.TrackBar2Change(Sender: TObject);
begin
  edLambdaDouble.Text:=IntToStr(TrackBar2.Position);
end;

end.
```

Lampiran 5 : Kode Program Difraksi Feynnman

```
(* Content-type: application/mathematica *)

(*** Wolfram Notebook File ***)
(* http://www.wolfram.com/nb *)

(* CreatedBy='Mathematica 7.0' *)

(*CacheID: 234*)
(* Internal cache information:
NotebookFileLineBreakTest
NotebookFileLineBreakTest
NotebookDataPosition[       145,          7]
NotebookDataLength[     24290,        643]
NotebookOptionsPosition[     22922,        593]
NotebookOutlinePosition[     23561,        617]
CellTagsIndexPosition[     23518,        614]
WindowTitle->Single-Slit Diffraction Using Feynman's Method - Source
WindowFrame->Normal*)

(* Beginning of Notebook Content *)
Notebook[{

Cell[CellGroupData[{
Cell["Single-Slit Diffraction Using Feynman's Method", "Section",
 CellFrameColor->RGBColor[
  0.6449835965514611, 0.758632791638056, 0.2516823071641108],
 FontColor->RGBColor[
  0.6449835965514611, 0.758632791638056, 0.2516823071641108]],

Cell[BoxData[
 RowBox[{"Manipulate", "[", "\[IndentingNewLine]",
  RowBox[{
   RowBox[{
    RowBox[{"If", "[",
     RowBox[{
      RowBox[{
       RowBox[{"n", ">", "20"}], "&&",
       RowBox[{"d", "===", "\"\<screen\>\""}]}]}], ",",
      RowBox[{"n", "=", "20"}]}]}], "]"}], ";", "\[IndentingNewLine]",
    RowBox[{"s", "=", "300"}], ";", "\[IndentingNewLine]",
    RowBox[{
     RowBox[{"\[CurlyPhi]", "[",
      RowBox[{"i_", ",", "x_"}], "]"}], ":=",
     RowBox[{
      FractionBox[
       RowBox[{
        SqrtBox[
         RowBox[{
          SuperscriptBox["s", "2"], "+",
          RowBox[{
```

```
            SuperscriptBox["b", "2"],
            SuperscriptBox[
             RowBox[{"(",
              RowBox[{
               FractionBox["1", "2"], " ", "-",
               FractionBox["i",
                RowBox[{"n", "+", "1"}]]}], ")"}], "2"]}]}]], "+",
          SqrtBox[
           RowBox[{
            SuperscriptBox["e", "2"], "+",
            SuperscriptBox[
             RowBox[{"(",
              RowBox[{"x", "-",
               RowBox[{"b",
                RowBox[{"(",
                 RowBox[{
                  FractionBox["1", "2"], " ", "-",
                  FractionBox["i",
                   RowBox[{"n", "+", "1"}]]}], ")"}]}]}], ")"}], "2"]}]]}],
        RowBox[{"\[Lambda]", " "}]],
       SuperscriptBox["10", "6"]}]}], ";", "\[IndentingNewLine]",
     RowBox[{
      RowBox[{"amp", "[", "x_", "]"}], ":=",
      RowBox[{
       SuperscriptBox[
        RowBox[{"(",
         RowBox[{
          UnderoverscriptBox["\[Sum]",
           RowBox[{"i", "=", "1"}], "n"],
          RowBox[{"Sin", "[",
           RowBox[{"2", "\[Pi]", " ",
            RowBox[{"\[CurlyPhi]", "[",
             RowBox[{"i", ",", "x"}], "]"}]}]}], "]"}]}], ")"}], "2"], "+",
       SuperscriptBox[
        RowBox[{"(",
         RowBox[{
          UnderoverscriptBox["\[Sum]",
           RowBox[{"i", "=", "1"}], "n"],
          RowBox[{"Cos", "[",
           RowBox[{"2", "\[Pi]", " ",
            RowBox[{"\[CurlyPhi]", "[",
             RowBox[{"i", ",", "x"}], "]"}]}]}], "]"}]}], ")"}], "2"]}]}],
";",
    "\[IndentingNewLine]",
    RowBox[{"Which", "[",
     RowBox[{
      RowBox[{"d", "===", "\"\<intensity\>\""}], ",",
      RowBox[{"Show", "[",
       RowBox[{
        RowBox[{"{",
         RowBox[{
          RowBox[{"ListLinePlot", "[",
           RowBox[{
            RowBox[{"Table", "[",
             RowBox[{
              RowBox[{"{",
               RowBox[{"j", ",",
                FractionBox[
                 RowBox[{"amp", "[", "j", "]"}],
                 SuperscriptBox["n", "2"]]}], "}"}], ",",
              RowBox[{"{",
               RowBox[{"j", ",",
```

```
                RowBox[{"-", "a"}], ",", "a", ",", "0.1"}], "}"}]}],
"]"}],
            ",",
            RowBox[{"AxesLabel", "\[Rule]",
             RowBox[{"{",
              RowBox[{
              "\"\<\!\(\*FractionBox[\(a\), \(mm\)]\)\>\"", ",",
               "\"\<I\>\""}], "}"}]}], ",",
            RowBox[{"PlotRange", "\[Rule]",
             RowBox[{"{",
              RowBox[{"0", ",", "1.1"}], "}"}]}]}], "]"}], ",",
          RowBox[{"Plot", "[",
           RowBox[{
            FractionBox[
             SuperscriptBox[
              RowBox[{"Sin", "[",
               RowBox[{
                FractionBox[
                 RowBox[{"\[Pi]", " ", "b"}], "\[Lambda]"],
                SuperscriptBox["10", "6"],
                RowBox[{"Sin", "[",
                 RowBox[{"ArcTan", "[",
                  FractionBox["x", "e"], "]"}], "]"}]}], "]"}], "2"],
             SuperscriptBox[
              RowBox[{"(",
               RowBox[{
                FractionBox[
                 RowBox[{"\[Pi]", " ", "b"}], "\[Lambda]"],
                SuperscriptBox["10", "6"],
                RowBox[{"Sin", "[",
                 RowBox[{"ArcTan", "[",
                  FractionBox["x", "e"], "]"}], "]"}]}], ")"}], "2"]],
",",
            RowBox[{"{",
             RowBox[{"x", ",",
              RowBox[{"-", "a"}], ",", "a"}], "}"}], ",",
            RowBox[{"PlotRange", "\[Rule]",
             RowBox[{"{",
              RowBox[{"0", ",", "1.1"}], "}"}]}], ",",
            RowBox[{"PlotStyle", "\[Rule]",
             RowBox[{"{",
              RowBox[{"Red", ",",
               RowBox[{"If", "[",
                RowBox[{"te", ",",
                 RowBox[{"Opacity", "[", "1", "]"}], ",",
                 RowBox[{"Opacity", "[", "0", "]"}]}], "]"}]}],
"}"}]}]}]}],
           "]"}]}], "}"}], ",",
         RowBox[{"ImageSize", "\[Rule]",
          RowBox[{"{",
           RowBox[{"350", ",", "300"}], "}"}]}]}], "]"}], ",",
       RowBox[{"d", "===", "\"\<screen\>\""}], ",",
       RowBox[{"ListDensityPlot", "[",
        RowBox[{
         RowBox[{"Flatten", "[",
          RowBox[{
           RowBox[{"Table", "[",
            RowBox[{
             RowBox[{"Table", "[",
              RowBox[{
               RowBox[{"{",
                RowBox[{"j", ",", "i", ",",
                 FractionBox[
```

```
                RowBox[{"amp", "[", "j", "]"}],
                SuperscriptBox["n", "2"]}]}], "}"}], ",",
              RowBox[{"{",
               RowBox[{"j", ",",
                RowBox[{"-", "a"}], ",", "a", ",", "0.1"}], "}"}]}]}],
"]"}],
            ",",
            RowBox[{"{",
             RowBox[{"i", ",", "1", ",", "10"}], "}"}]}], "]"}], ",",
"1"}],
         "]"}], ",",
        RowBox[{"FrameTicks", "\[Rule]",
         RowBox[{"{",
          RowBox[{"Automatic", ",", "None"}], "}"}]}], ",",
        RowBox[{"InterpolationOrder", "\[Rule]", "3"}], ",",
        RowBox[{"PlotRange", "\[Rule]", "All"}], ",",
        RowBox[{"ColorFunction", "\[Rule]", "GrayLevel"}], ",",
        RowBox[{"ColorFunctionScaling", "\[Rule]", "False"}], ",",
        RowBox[{"ImageSize", "\[Rule]",
         RowBox[{"{",
          RowBox[{"350", ",", "300"}], "}"}]}]}], "]"}]}], "]"}]}], ",",
   "\[IndentingNewLine]",
   RowBox[{"{",
    RowBox[{
     RowBox[{"{",
      RowBox[{
       "e", ",", "300", ",", "\"\<distance of slit\\nto screen
(mm)\>\""}],
       "}"}], ",", "300", ",", "1000", ",", "100", ",",
     RowBox[{"Appearance", "\[Rule]", "\"\<Labeled\>\""}], ",",
     RowBox[{"ImageSize", "\[Rule]", "Tiny"}]}], "}"}], ",",
   "\[IndentingNewLine]",
   RowBox[{"{",
    RowBox[{
     RowBox[{"{",
      RowBox[{"b", ",", "0.05", ",", "\"\<width of slit (mm)\>\""}],
"}"}],
     ",", "0.01", ",", "0.5", ",", "0.01", ",",
     RowBox[{"Appearance", "\[Rule]", "\"\<Labeled\>\""}], ",",
     RowBox[{"ImageSize", "\[Rule]", "Tiny"}]}], "}"}], ",",
   RowBox[{"{",
    RowBox[{
     RowBox[{"{",
      RowBox[{"\[Lambda]", ",", "650", ",", "\"\<wavelength (nm)\>\""}],
      "}"}], ",", "450", ",", "750", ",", "1", ",",
     RowBox[{"Appearance", "\[Rule]", "\"\<Labeled\>\""}], ",",
     RowBox[{"ImageSize", "\[Rule]", "Tiny"}]}], "}"}], ",",
   "\[IndentingNewLine]",
   RowBox[{"{",
    RowBox[{
     RowBox[{"{",
      RowBox[{"a", ",", "8", ",", "\"\<width of image (mm)\>\""}], "}"}],
",",
      "2", ",", "8", ",", "1", ",",
     RowBox[{"Appearance", "\[Rule]", "\"\<Labeled\>\""}], ",",
     RowBox[{"ImageSize", "\[Rule]", "Tiny"}]}], "}"}], ",",
   RowBox[{"{",
    RowBox[{
     RowBox[{"{",
      RowBox[{"n", ",", "20", ",", "\"\<number of paths\>\""}], "}"}],
",",
     "10", ",",
     RowBox[{"Dynamic", "[",
```

```
      RowBox[{"If", "[",
       RowBox[{
        RowBox[{"d", "===", "\"\<intensity\>\""}], ",", "100", ",",
"20"}],
       "]"}], "]"}], ",", "10", ",",
     RowBox[{"Appearance", "\[Rule]", "\"\<Labeled\>\""}], ",",
     RowBox[{"ImageSize", "\[Rule]", "Tiny"}]}], "}"}], ",",
   "\[IndentingNewLine]", "Delimiter", ",", "\[IndentingNewLine]",
   RowBox[{"{",
    RowBox[{
     RowBox[{"{",
      RowBox[{"d", ",", "\"\<intensity\>\"", ",", "\"\<image\>\""}],
"}"}],
     ",",
     RowBox[{"{",
      RowBox[{"\"\<intensity\>\"", ",", "\"\<screen\>\""}], "}"}]}],
"}"}],
   ",", "\[IndentingNewLine]",
   RowBox[{"{",
    RowBox[{
     RowBox[{"{",
      RowBox[{"te", ",", "False", ",", "\"\<theoretical curve\>\""}],
"}"}],
     ",",
     RowBox[{"{",
      RowBox[{"True", ",", "False"}], "}"}], ",",
     RowBox[{"Enabled", "\[Rule]",
      RowBox[{"Dynamic", "[",
       RowBox[{"d", "===", "\"\<intensity\>\""}], "]"}]}]}], "}"}], ",",
   "\[IndentingNewLine]",
   RowBox[{"SynchronousUpdating", "\[Rule]", "False"}], ",",
   "\[IndentingNewLine]",
   RowBox[{"TrackedSymbols", "\[Rule]", "True"}], ",",
   RowBox[{"ControlPlacement", "\[Rule]", "Left"}], ",",
   RowBox[{"AutorunSequencing", "\[Rule]",
    RowBox[{"{",
     RowBox[{"1", ",", "2", ",", "3", ",", "4", ",", "6", ",", "7"}],
     "}"}]}]}], "]"}]}], "Input",
 CellChangeTimes->{
  3.35696210375764*^9, {3.442988055748375*^9, 3.442988240213385*^9}, {
   3.442988297754942*^9, 3.442988301302553*^9}, {3.442988449211864*^9,
   3.442988469410698*^9}, {3.442988596726787*^9, 3.442988601478222*^9},
   3.4429892914455233`*^9, {3.443004666079076*^9, 3.443004667318425*^9},
   3.443005745951708*^9, {3.443291524810502*^9, 3.4432916945834775`*^9},
{
   3.443300782173944*^9, 3.4433008047520695`*^9}, 3.443300904892694*^9, {
   3.4433010765958195`*^9, 3.443301080298944*^9}, {3.443301131892694*^9,
   3.4433011370333195`*^9}, {3.44337998929624*^9,
3.4433800058098907`*^9}, {
   3.4433801434536533`*^9, 3.443380198413149*^9}, {3.443380241016409*^9,
   3.443380318715178*^9}, {3.469460630428316*^9, 3.469460630554329*^9}}]
}, Open  ]],

Cell[CellGroupData[{

Cell[BoxData[
 TagBox[
  StyleBox[
   DynamicModuleBox[{$CellContext`a$$ = 8, $CellContext`b$$ =
    0.05, $CellContext`d$$ = "intensity", $CellContext`e$$ =
    300, $CellContext`n$$ = 20, $CellContext`te$$ =
    False, $CellContext`\[Lambda]$$ = 650, Typeset`show$$ = True,
    Typeset`bookmarkList$$ = {}, Typeset`bookmarkMode$$ = "Menu",
```

```
    Typeset`animator$$, Typeset`animvar$$ = 1, Typeset`name$$ =
    "\"untitled\"", Typeset`specs$$ = {{{
       Hold[$CellContext`e$$], 300, "distance of slit\nto screen (mm)"},
300,
      1000, 100}, {{
       Hold[$CellContext`b$$], 0.05, "width of slit (mm)"}, 0.01, 0.5,
      0.01}, {{
       Hold[$CellContext`\[Lambda]$$], 650, "wavelength (nm)"}, 450, 750,
      1}, {{
       Hold[$CellContext`a$$], 8, "width of image (mm)"}, 2, 8, 1}, {{
       Hold[$CellContext`n$$], 20, "number of paths"}, 10,
      Dynamic[
       If[$CellContext`d$$ === "intensity", 100, 20]], 10}, {{
       Hold[$CellContext`d$$], "intensity", "image"}, {
      "intensity", "screen"}}, {{
       Hold[$CellContext`te$$], False, "theoretical curve"}, {True,
False}}},
    Typeset`size$$ = {350., {148., 152.}}, Typeset`update$$ = 0,
    Typeset`initDone$$, Typeset`skipInitDone$$ =
    True, $CellContext`e$25255$$ = 0, $CellContext`b$25256$$ =
    0, $CellContext`\[Lambda]$25257$$ = 0, $CellContext`a$25258$$ =
    0, $CellContext`n$25259$$ = 0, $CellContext`d$25260$$ =
    False, $CellContext`te$25261$$ = False},
    DynamicBox[Manipulate`ManipulateBoxes[
     1, StandardForm,
      "Variables" :> {$CellContext`a$$ = 8, $CellContext`b$$ =
        0.05, $CellContext`d$$ = "intensity", $CellContext`e$$ =
        300, $CellContext`n$$ = 20, $CellContext`te$$ =
        False, $CellContext`\[Lambda]$$ = 650}, "ControllerVariables" :>
{
        Hold[$CellContext`e$$, $CellContext`e$25255$$, 0],
        Hold[$CellContext`b$$, $CellContext`b$25256$$, 0],
        Hold[$CellContext`\[Lambda]$$, $CellContext`\[Lambda]$25257$$,
0],
        Hold[$CellContext`a$$, $CellContext`a$25258$$, 0],
        Hold[$CellContext`n$$, $CellContext`n$25259$$, 0],
        Hold[$CellContext`d$$, $CellContext`d$25260$$, False],
        Hold[$CellContext`te$$, $CellContext`te$25261$$, False]},
      "OtherVariables" :> {
       Typeset`show$$, Typeset`bookmarkList$$, Typeset`bookmarkMode$$,
        Typeset`animator$$, Typeset`animvar$$, Typeset`name$$,
        Typeset`specs$$, Typeset`size$$, Typeset`update$$,
Typeset`initDone$$,
         Typeset`skipInitDone$$}, "Body" :> (If[
         And[$CellContext`n$$ > 20, $CellContext`d$$ ===
          "screen"], $CellContext`n$$ = 20]; $CellContext`s =
        300; $CellContext`\[CurlyPhi][
          Pattern[$CellContext`i$,
           Blank[]],
          Pattern[$CellContext`x$,

           Blank[]]] := ((($CellContext`s^2 + $CellContext`b$$^2 (
                1/2 - $CellContext`i$/($CellContext`n$$ + 1))^2)^
            Rational[
             1, 2] + ($CellContext`e$$^2 + ($CellContext`x$ - \
$CellContext`b$$ (1/2 - $CellContext`i$/($CellContext`n$$ + 1)))^2)^
           Rational[1, 2])/$CellContext`\[Lambda]$$) 10^6;
$CellContext`amp[
          Pattern[$CellContext`x$,
           Blank[]]] := Sum[
            Sin[
            2 Pi $CellContext`\[CurlyPhi][$CellContext`i,
$CellContext`x$]], \
```

```
{$CellContext`i, 1, $CellContext`n$$}]^2 + Sum[
            Cos[
            2 Pi $CellContext`\[CurlyPhi][$CellContext`i,
$CellContext`x$]], \
{$CellContext`i, 1, $CellContext`n$$}]^2;
       Which[$CellContext`d$$ === "intensity",
         Show[{
           ListLinePlot[

            Table[{$CellContext`j, \
$CellContext`amp[$CellContext`j]/$CellContext`n$$^2}, {$CellContext`j, \
-$CellContext`a$$, $CellContext`a$$, 0.1}],
            AxesLabel -> {"\!\(\*FractionBox[\(a\), \(mm\)]\)", "I"},
            PlotRange -> {0, 1.1}],
           Plot[Sin[(Pi $CellContext`b$$/$CellContext`\[Lambda]$$) 10^6
Sin[
                 ArcTan[$CellContext`x/$CellContext`e$$]]]^2/((
              Pi $CellContext`b$$/$CellContext`\[Lambda]$$) 10^6 Sin[

               ArcTan[$CellContext`x/$CellContext`e$$]])^2,
{$CellContext`x, \
-$CellContext`a$$, $CellContext`a$$}, PlotRange -> {0, 1.1},
            PlotStyle -> {Red,
              If[$CellContext`te$$,
               Opacity[1],
               Opacity[0]]}]}, ImageSize -> {350, 300}], $CellContext`d$$
===
         "screen",
         ListDensityPlot[
          Flatten[
           Table[

            Table[{$CellContext`j, $CellContext`i, \
$CellContext`amp[$CellContext`j]/$CellContext`n$$^2}, {$CellContext`j, \
-$CellContext`a$$, $CellContext`a$$, 0.1}], {$CellContext`i, 1, 10}], 1],
          FrameTicks -> {Automatic, None}, InterpolationOrder -> 3,
PlotRange ->
          All, ColorFunction -> GrayLevel, ColorFunctionScaling -> False,
          ImageSize -> {350, 300}]]),
      "Specifications" :> {{{$CellContext`e$$, 300,
          "distance of slit\nto screen (mm)"}, 300, 1000, 100, Appearance
->
         "Labeled", ImageSize ->
         Tiny}, {{$CellContext`b$$, 0.05, "width of slit (mm)"}, 0.01,
0.5,
         0.01, Appearance -> "Labeled", ImageSize ->
         Tiny}, {{$CellContext`\[Lambda]$$, 650, "wavelength (nm)"}, 450,
750,
          1, Appearance -> "Labeled", ImageSize ->
         Tiny}, {{$CellContext`a$$, 8, "width of image (mm)"}, 2, 8, 1,
         Appearance -> "Labeled", ImageSize ->
         Tiny}, {{$CellContext`n$$, 20, "number of paths"}, 10,
         Dynamic[
          If[$CellContext`d$$ === "intensity", 100, 20]], 10, Appearance
->
         "Labeled", ImageSize -> Tiny},
        Delimiter, {{$CellContext`d$$, "intensity", "image"}, {
         "intensity", "screen"}}, {{$CellContext`te$$, False,
          "theoretical curve"}, {True, False}, Enabled ->
         Dynamic[$CellContext`d$$ === "intensity"]}},
      "Options" :> {
       SynchronousUpdating -> False, TrackedSymbols -> True,
ControlPlacement ->
```

```
        Left, AutorunSequencing -> {1, 2, 3, 4, 6, 7}},
       "DefaultOptions" :> {ControllerLinking -> True}],
      ImageSizeCache->{616., {177., 182.}},
      SingleEvaluation->True],
     Deinitialization:>None,
     DynamicModuleValues:>{},
     SynchronousInitialization->True,
     UnsavedVariables:>{Typeset`initDone$$},
     UntrackedVariables:>{Typeset`size$$}], "Manipulate",
    Deployed->True,
    StripOnInput->False],
   Manipulate`InterpretManipulate[1]]], "Output",
  CellID->815146555],

Cell[CellGroupData[{

Cell["Caption", "Section",
 CellFrameColor->RGBColor[
  0.6449835965514611, 0.758632791638056, 0.2516823071641108],
 FontColor->RGBColor[
  0.6449835965514611, 0.758632791638056, 0.2516823071641108]],

Cell["\<\
This Demonstration investigates how the different parameters influence
the \
image of the diffraction. The calculation of the intensity is based on \
Feynman's method of \"integrating over paths\". You can observe how the \
number of paths changes the intensity and can compare it with the
theoretical \
curve of Fraunhofer diffraction. For a large aperture, for example 0.5
mm, \
there are large differences between the calculated curve and the
theoretical \
curve, because the diffraction is not a Fraunhofer but a Fresnel
diffraction. \
The image can show the graph of the intensity or a density plot that \
resembles the image seen on a screen in a real experiment. For this
setting \
the maximum number of paths is limited to 20 in order to get a faster \
response.\
\>", "Text"]
}, Close]]
}, Open  ]],
Cell[CellGroupData[{
Cell["Details", "Section",
 CellFrameColor->RGBColor[
  0.6449835965514611, 0.758632791638056, 0.2516823071641108],
 FontColor->RGBColor[
  0.6449835965514611, 0.758632791638056, 0.2516823071641108]],

Cell[TextData[{
 "R. P. Feynman, ",
 StyleBox["QED: The Strange Theory of Light and Matter",
  FontSlant->"Italic"],
 ", Princeton: Princeton University Press, 1985."
}], "Text"]
}, Close]],

Cell[TextData[{
 "R. P. Feynman, ",
 StyleBox["QED: The Strange Theory of Light and Matter",
  FontSlant->"Italic"],
 ", Princeton: Princeton University Press, 1985."
```

```
}], "Text"],
Cell[CellGroupData[{
Cell["THIS NOTEBOOK IS THE SOURCE CODE FROM", "Text",
 CellFrame->{{0, 0}, {0, 1}},
 CellMargins->{{48, 10}, {4, 28}},
 CellGroupingRules->{"SectionGrouping", 25},
 CellFrameMargins->{{48, 48}, {6, 5}},
 CellFrameColor->RGBColor[0.691905, 0.790311, 0.300252],
 FontFamily->"Helvetica",
 FontSize->10,
 FontWeight->"Bold",
 FontColor->RGBColor[0.691905, 0.790311, 0.300252]],
Cell[TextData[{
 "\"",
 ButtonBox["Single-Slit Diffraction Using Feynman's Method",
  BaseStyle->"Hyperlink",
  ButtonData->{
    URL["http://demonstrations.wolfram.com/\
SingleSlitDiffractionUsingFeynmansMethod/"], None},
  ButtonNote->
   "http://demonstrations.wolfram.com/\
SingleSlitDiffractionUsingFeynmansMethod/"],
 "\"",
 " from ",
 ButtonBox["The Wolfram Demonstrations Project",
  BaseStyle->"Hyperlink",
  ButtonData->{
    URL["http://demonstrations.wolfram.com/"], None},
  ButtonNote->"http://demonstrations.wolfram.com/"],
 "\[ParagraphSeparator]\[NonBreakingSpace]",
 ButtonBox["http://demonstrations.wolfram.com/\
SingleSlitDiffractionUsingFeynmansMethod/",
  BaseStyle->"Hyperlink",
  ButtonData->{
    URL["http://demonstrations.wolfram.com/\
SingleSlitDiffractionUsingFeynmansMethod/"], None},
  ButtonNote->
   "http://demonstrations.wolfram.com/\
SingleSlitDiffractionUsingFeynmansMethod/"]
}], "Text",
 CellMargins->{{48, Inherited}, {0, Inherited}},
 FontFamily->"Verdana",
 FontSize->10,
 FontColor->GrayLevel[0.5]],
Cell[" ", "Text",
 CellFrame->{{0, 0}, {0, 1}},
 CellMargins->{{48, 10}, {4, 28}},
 CellGroupingRules->{"SectionGrouping", 25},
 CellFrameMargins->{{48, 48}, {6, 5}},
 CellFrameColor->RGBColor[0.691905, 0.790311, 0.300252],
 FontFamily->"Helvetica",
 FontSize->10,
 FontWeight->"Bold",
 FontColor->RGBColor[0.691905, 0.790311, 0.300252]],

Cell[TextData[{
 "Contributed by: ",
 ButtonBox["Hans-Joachim Domke",
  BaseStyle->"Hyperlink",
  ButtonData->{
    URL["http://demonstrations.wolfram.com/author.html?author=Hans-
Joachim+\
Domke"], None},
```

```
  ButtonNote->
   "http://demonstrations.wolfram.com/author.html?author=Hans-
Joachim+Domke"]
}], "Text",
 CellDingbat->"\[FilledSmallSquare]",
 CellMargins->{{66, 48}, {2, 4}},
 FontFamily->"Verdana",
 FontSize->10,
 FontColor->GrayLevel[0.6]],

Cell[CellGroupData[{
Cell[TextData[{
 "A full-function Wolfram ",
 StyleBox["Mathematica",
  FontSlant->"Italic"],
 " system (Version 6 or higher) is required to edit this notebook.\n",
 StyleBox[ButtonBox["GET WOLFRAM MATHEMATICA \[RightGuillemet]",
  BaseStyle->"Hyperlink",
  ButtonData->{
    URL["http://www.wolfram.com/products/mathematica/"], None},
  ButtonNote->"http://www.wolfram.com/products/mathematica/"],
  FontFamily->"Helvetica",
  FontWeight->"Bold",
  FontSlant->"Italic",
  FontColor->RGBColor[1, 0.42, 0]]
}], "Text",
 CellFrame->True,
 CellMargins->{{48, 68}, {8, 28}},
 CellFrameMargins->12,
 CellFrameColor->RGBColor[0.865507, 0.90634, 0.680751],
 CellChangeTimes->{3.3750111182355957`*^9},
 ParagraphSpacing->{1., 1.},
 FontFamily->"Verdana",
 FontSize->10,
 FontColor->GrayLevel[0.411765],
 Background->RGBColor[0.986023, 0.991363, 0.969818]],

Cell[TextData[{
 "\[Copyright] ",
 StyleBox[ButtonBox["The Wolfram Demonstrations Project & Contributors",
  BaseStyle->"Hyperlink",
  ButtonData->{
    URL["http://demonstrations.wolfram.com/"], None},
  ButtonNote->"http://demonstrations.wolfram.com/"],
  FontColor->GrayLevel[0.6]],
 "\[ThickSpace]\[ThickSpace]\[ThickSpace]|\[ThickSpace]\[ThickSpace]\
\[ThickSpace]",
 StyleBox[ButtonBox["Terms of Use",
  BaseStyle->"Hyperlink",
  ButtonData->{
    URL["http://demonstrations.wolfram.com/termsofuse.html"], None},
  ButtonNote->"http://demonstrations.wolfram.com/termsofuse.html"],
  FontColor->GrayLevel[0.6]],
 "\[ThickSpace]\[ThickSpace]\[ThickSpace]|\[ThickSpace]\[ThickSpace]\
\[ThickSpace]",
 StyleBox[ButtonBox["Make a new version of this Demonstration \
\[RightGuillemet]",
  BaseStyle->"Hyperlink",
  ButtonData->{
    URL["http://demonstrations.wolfram.com/participate/upload.jsp?id=\
SingleSlitDiffractionUsingFeynmansMethod"], None},
  ButtonNote->None],
  FontColor->GrayLevel[0.6]]
```

```
}], "Text",
 CellFrame->{{0, 0}, {0, 0.5}},
 CellMargins->{{48, 10}, {20, 50}},
 CellFrameMargins->{{6, 0}, {6, 6}},
 CellFrameColor->GrayLevel[0.6],
 FontFamily->"Verdana",
 FontSize->9,
 FontColor->GrayLevel[0.6]]
}, Open  ]]
}, Open  ]]
},
Editable->True,
Saveable->False,
ScreenStyleEnvironment->"Working",
WindowSize->{780, 650},
WindowMargins->{{Inherited, Inherited}, {Inherited, 0}},
WindowElements->{
 "StatusArea", "MemoryMonitor", "MagnificationPopUp",
"VerticalScrollBar",
  "MenuBar"},
WindowTitle->"Single-Slit Diffraction Using Feynman's Method - Source",
DockedCells->{},
CellContext->Notebook,
FrontEndVersion->"7.0 for Microsoft Windows (32-bit) (February 18,
2009)",
StyleDefinitions->"Default.nb"
]
(* End of Notebook Content *)

(* Internal cache information *)
(*CellTagsOutline
CellTagsIndex->{}
*)
(*CellTagsIndex
CellTagsIndex->{}
*)
(*NotebookFileOutline
Notebook[{
Cell[CellGroupData[{
Cell[636, 23, 238, 4, 70, "Section"],
Cell[877, 29, 9516, 243, 70, "Input"]
}, Open  ]],
Cell[CellGroupData[{
Cell[10430, 277, 6499, 129, 70, "Output",
 CellID->815146555],
Cell[CellGroupData[{
Cell[16954, 410, 199, 4, 70, "Section"],
Cell[17156, 416, 794, 12, 70, "Text"]
}, Close]]
}, Open  ]],
Cell[CellGroupData[{
Cell[17998, 434, 199, 4, 70, "Section"],
Cell[18200, 440, 178, 5, 70, "Text"]
}, Close]],
Cell[18392, 448, 178, 5, 70, "Text"],
Cell[CellGroupData[{
Cell[18595, 457, 373, 9, 70, "Text",
 CellGroupingRules->{"SectionGrouping", 25}],
Cell[18971, 468, 1041, 31, 70, "Text"],
Cell[20015, 501, 337, 9, 70, "Text",
 CellGroupingRules->{"SectionGrouping", 25}],
Cell[20355, 512, 448, 14, 70, "Text"],
Cell[CellGroupData[{
```

```
Cell[20828, 530, 847, 24, 70, "Text"],
Cell[21678, 556, 1216, 33, 70, "Text"]
}, Open  ]]
}, Open  ]]
}
]
*)
(* End of internal cache information *)
(* NotebookSignature ATjWFgydtLp3qCpzk8d6460d *)
```

Lampiran 6 : Kode Program Difraksi Celah Simetris (*Square Apperture* dan *Circular Apperture*)

```
unit Difraksi;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
Forms,
  Dialogs, Grids, StdCtrls, ComCtrls, TeEngine, Series, ExtCtrls,
TeeProcs,
  Chart,jpeg, ExtDlgs;

type
  TForm1 = class(TForm)
    PageControl1: TPageControl;
    TabSheet1: TTabSheet;
    TabSheet2: TTabSheet;
    PageControl2: TPageControl;
    TabSheet3: TTabSheet;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Label4: TLabel;
    Label5: TLabel;
    Edit1: TEdit;
    Edit2: TEdit;
    Edit3: TEdit;
    Edit4: TEdit;
    Edit5: TEdit;
    TabSheet4: TTabSheet;
    Button1: TButton;
    StringGrid1: TStringGrid;
    StringGrid2: TStringGrid;
    StringGrid3: TStringGrid;
    Chart1: TChart;
    Series1: TFastLineSeries;
    Button2: TButton;
    Label6: TLabel;
    Label7: TLabel;
    Label8: TLabel;
    Button3: TButton;
    Button4: TButton;
    ProgressBar1: TProgressBar;
    Button5: TButton;
    Button6: TButton;
    Button7: TButton;
    Button8: TButton;
    SaveDialog1: TSaveDialog;
    Button9: TButton;
```

```
    SavePictureDialog1: TSavePictureDialog;
    Series3: TFastLineSeries;
    Chart3: TChart;
    Series2: TFastLineSeries;
    TabSheet5: TTabSheet;
    Memo1: TMemo;
    Button10: TButton;
    I_celah: TImage;
    Button11: TButton;
    I_Layar: TImage;
    Chart2: TChart;
    Series4: TFastLineSeries;
    Memo2: TMemo;
    Button12: TButton;
    Button13: TButton;
    Button14: TButton;
    GroupBox1: TGroupBox;
    RadioButton1: TRadioButton;
    RadioButton2: TRadioButton;
    ProgressBar2: TProgressBar;
    procedure Button1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
    procedure Button3Click(Sender: TObject);
    procedure Button4Click(Sender: TObject);
    procedure Button5Click(Sender: TObject);
    procedure Button6Click(Sender: TObject);
    procedure Button7Click(Sender: TObject);
    procedure Button8Click(Sender: TObject);
    procedure Button9Click(Sender: TObject);
    procedure Button10Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure FormClose(Sender: TObject; var Action: TCloseAction);
    procedure I_celahMouseDown(Sender: TObject; Button: TMouseButton;
      Shift: TShiftState; X, Y: Integer);
    procedure Button11Click(Sender: TObject);
    procedure Button12Click(Sender: TObject);
    procedure Button13Click(Sender: TObject);
    procedure Button14Click(Sender: TObject);
    procedure RadioButton1Click(Sender: TObject);
    procedure RadioButton2Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;
  JumlahBerkas,JarakAntarPoin,JarakCelahKeLayar,Layar:integer;
 A,B,C,D,E,F,G,H,P,Q,W,X:array[-1000..1000,-1000..1000]of extended;
 i,j,k,l,m,n,r,s,pb: integer;
 BilGelombang,Amplitudo,PanjangGelombang,delta,hasil:double;
  BMP_Celah,BMP_Layar:TBitmap;
  jumlahTitik:integer;
  titik:array[1..50] of TPoint;

implementation

{$R *.dfm}

procedure TForm1.Button1Click(Sender: TObject);
var
ncol:integer;
```

```
begin
Memo1.Lines.Clear;
Label6.Caption:=('Pitagoras');
Label7.Caption:=('Selisih Jarak');
Label8.Caption:=('Delta');
Label7.Visible:=True;
Label8.Visible:=True;
StringGrid2.Visible:=True;
StringGrid3.Visible:=True;
Chart3.Visible:=False;
StringGrid1.RowCount:=2;

//reset hasil
for i:=-k to k do
                begin
                        for j:=-l to l do
                        begin
                        A[i,j]:=0;
                        B[i,j]:=0;
                        C[i,j]:=0;
                        D[i,j]:=0;
                        E[i,j]:=0;
                        F[i,j]:=0;
                        G[i,j]:=0;
                        H[i,j]:=0;
                        P[i,j]:=0;
                        Q[i,j]:=0;
                        W[i,j]:=0;
                        X[i,j]:=0;
                        end;

                end;


for ncol:=0 to StringGrid1.ColCount-1 do
StringGrid1.Cols[ncol].Clear;
//Memo1.Lines.Clear;

StringGrid2.RowCount:=2;
for ncol:=0 to StringGrid2.ColCount-1 do
StringGrid2.Cols[ncol].Clear;

StringGrid3.RowCount:=2;
for ncol:=0 to StringGrid3.ColCount-1 do
StringGrid3.Cols[ncol].Clear;

//mulai memasukkan data
  JumlahBerkas := StrToInt(Edit1.Text);
  JarakAntarPoin := StrToInt(Edit2.Text);
  JarakCelahKeLayar := StrToInt(Edit3.Text);
  PanjangGelombang:= StrToFloat(Edit4.Text);
  Amplitudo:= StrToFloat(Edit5.Text);
  BilGelombang:= ((JumlahBerkas*pi)/PanjangGelombang);

   Layar:=JumlahBerkas*20;
  //syarat batas iterasi
  k:= Layar div 2;
  l:= JumlahBerkas div 2;
  //pembedaan ganjil dan genap
  m:= Layar mod 2;
  n:= JumlahBerkas mod 2;
```

```
 //mengatur tampilan sringgrid.
 StringGrid1.ColCount:=(2*k)+2;
 StringGrid1.RowCount:=(2*l)+2;
  //memulai iterasi
  for i:=-k to k do
  begin
  StringGrid1.Cells[i+k+1,0]:=IntToStr(i);
        for j:=-l to l do
        begin
        StringGrid1.Cells[0,j+l+1]:=IntToStr(j);
          if (i=j) then
          A[i,j]:=JarakCelahKeLayar
          else
          A[i,j]:=sqrt(sqr(JarakCelahKeLayar) + sqr(j - i));
          //if (m=0) then   <<<----- apa ini ya....????? huhuhuhuhu gak
paham
          //A[0,j]:=0;
          end;
       ProgressBar1.Position:=i;
  end;
  //menampilkan ke stringgrid
                for i:=-k to k do
                begin
                        for j:=-l to l do
                        begin

StringGrid1.Cells[i+k+1,j+l+1]:=FloatToStr(A[i,j]);
                        Memo1.Lines.Add(IntToStr(i)+ '
'+FloatToStr(A[i,j]));
                        end;

                end;
                StringGrid2.ColCount:=(2*k)+2;
                StringGrid2.RowCount:=(2*l)+2;

                StringGrid3.ColCount:=(2*k)+2;
                StringGrid3.RowCount:=(2*l)+2;
                for i:=-k to k do
                begin           // mencari selisih jarak
                StringGrid2.Cells[i+k+1,0]:=IntToStr(i);
                StringGrid3.Cells[i+k+1,0]:=IntToStr(i);
                //hasil:=0;
                  for j:=-l to (l-1) do
                        begin
                     StringGrid2.Cells[0,j+l+1]:=IntToStr(j);
                     StringGrid3.Cells[0,j+l+1]:=IntToStr(j);
                       H[i,j]:=A[i,j+1]-A[i,-l];

StringGrid2.Cells[i+k+1,j+l+1]:=FloatToStr(H[i,j]);

                    //mencari delta(k*Selisih jarak)
                    P[i,j]:=(BilGelombang*H[i,j]);
                    P[i,j]:=P[i,l-1];
                    //P[i,j]:=delta;
                    StringGrid3.Cells[i+k+1,j+l+1]:=FloatToStr(P[i,j]);
                        end;
                end;

    //                   end;

end;

procedure TForm1.Button2Click(Sender: TObject);
```

```
var
ncol:integer;
begin
Label6.Caption:=('Simpangan Real');
Label7.Caption:=('Simpangan Imaginer');
Label8.Caption:=('Delta');
Label7.Visible:=True;
Label8.Visible:=True;
StringGrid2.Visible:=True;
StringGrid3.Visible:=True;
Chart3.Visible:=False;
Memo1.Clear;
//mengatur tampilan sringgrid.
 StringGrid1.ColCount:=(2*k)+2;
 StringGrid1.RowCount:=(2*l)+2;
 StringGrid2.ColCount:=(2*k)+2;
 StringGrid2.RowCount:=(2*l)+2;
 StringGrid3.RowCount:=(2*k)+2;
 StringGrid3.RowCount:=(2*l)+2;
 //peramater gelombang

StringGrid3.RowCount:=2;
for ncol:=0 to StringGrid3.ColCount-1 do
StringGrid3.Cols[ncol].Clear;

              for i:=-k to k do
              begin
              StringGrid1.Cells[i+k+1,0]:=IntToStr(i);
              StringGrid2.Cells[i+k+1,0]:=IntToStr(i);
              StringGrid3.Cells[i+k+1,0]:=IntToStr(i);
              hasil:=0;
                     for j:=-l to l do
                     begin   // i=syarat batas
                     StringGrid1.Cells[0,j+l+1]:=IntToStr(j);

B[i,j]:=Amplitudo*cos(BilGelombang*(A[i,j]));//*(pi/180);
                     //-------------------- harus diubah dulu ke
radian

StringGrid1.Cells[i+k+1,j+l+1]:=FloatToStr(B[i,j]);
                     //--------------------   (+) donk.....
                     StringGrid2.Cells[0,j+l+1]:=IntToStr(j);

E[i,j]:=Amplitudo*sin(BilGelombang*(A[i,j]));//*(pi/180);

StringGrid2.Cells[i+k+1,j+l+1]:=FloatToStr(E[i,j]);
                     //ngitung total delta.....
                     StringGrid3.Cells[0,j+l+1]:=IntToStr(j);
                     //P[i,j]:=(BilGelombang*H[i,j]);
                     hasil:=hasil+P[i,j];
                     end;
                     Q[i,1]:=hasil;
                     StringGrid3.Cells[i+k+1,1]:=FloatToStr (Q[i,1]);
                     Memo1.Lines.Add(IntToStr(i)+'
'+FloatToStr(Q[i,1]));
                     //huhuhuhuhu aq gak mudeng..... (_ _")
                     end;
                     ProgressBar1.Position:=2*i;

end;

procedure TForm1.Button3Click(Sender: TObject);
var
```

```
nilai1,nilai2,nilai : double;
f,g: integer;
ncol:integer;
begin
Label6.Caption:=('Superposisi Real');
Label7.Caption:=('Superposisi Imaginer');
Label8.Caption:=('Delta');
Label7.Visible:=True;
Label8.Visible:=True;
Chart3.Visible:=False;
StringGrid2.Visible:=True;
StringGrid3.Visible:=True;
Series1.Clear; //ngitung superposisi gelombang Real & imaginer
Series2.Clear;
Memo1.Lines.Clear;
//f:=(2*k)+2;
//g:=(2*i)+2;
//nilai:=0;
//------------------------------ ngresiki stringgrid !!!!
StringGrid1.RowCount:=2;
for ncol:=0 to StringGrid1.ColCount-1 do
StringGrid1.Cols[ncol].Clear;
               for i:=-k to k do
               begin
               StringGrid1.Cells[i+k+1,0]:=IntToStr(i);
               nilai:=0;
                       for j:=-l to l do
                       begin
                       StringGrid1.Cells[0,j+l+1]:=IntToStr(j);
                       nilai:=nilai+B[i,j];
                       end;
               D[i,1]:=nilai;
               StringGrid1.Cells[i+k+1,1]:=FloatToStr(D[i,1]);
               end;
                   // piye carane ngitung superposisi... huhuhuhuhu
               StringGrid2.RowCount:=2;
               for ncol:=0 to StringGrid2.ColCount-1 do
               StringGrid2.Cols[ncol].Clear;
                for i:=-k to k do
               begin
               StringGrid2.Cells[i+k+1,0]:=IntToStr(i);
               nilai2:=0;
                       for j:=-l to l do
                       begin
                       StringGrid2.Cells[0,j+l+1]:=IntToStr(j);
                       nilai2:= nilai2 + E[i,j];
                       end;
               C[i,1]:=nilai2;
               StringGrid2.Cells[i+k+1,1]:=FloatToStr(C[i,1]);
               Memo1.Lines.Add(IntToStr(i)+'  '+FloatToStr(C[i,1]));
               Series1.AddXY(i,D[i,1]);
               Series2.AddXY(i,D[i,1]);
               ProgressBar1.Position:=3*i;
              end;


end;

procedure TForm1.Button4Click(Sender: TObject);
var
  nilai,nilai1,delta, intensitas : double;
  Layar,i,JumlahBerkas,lebar,s:integer;
  ncol,max:integer;
```

```
  Spectrum: TBitmap;
  r, g, b:byte;
begin         //ngitung intensitas
Label6.Caption:=('Intensitas Point');
Label7.Visible:=False;
Label8.Visible:=False;
StringGrid2.Visible:=False;
StringGrid3.Visible:=False;
Series1.Clear;
Series2.Clear;
Memo1.Lines.Clear;
StringGrid1.RowCount:=2;
for ncol:=0 to StringGrid1.ColCount-1 do
StringGrid1.Cols[ncol].Clear;

StringGrid2.RowCount:=2;
for ncol:=0 to StringGrid2.ColCount-1 do
StringGrid2.Cols[ncol].Clear;

StringGrid3.RowCount:=2;
for ncol:=0 to StringGrid3.ColCount-1 do
StringGrid3.Cols[ncol].Clear;

begin
  for i:=-k to k do
    begin
    StringGrid1.Cells[i+k+1,0]:=IntToStr(i);
    delta:=0;
    for j:=-l to l do       // piye iki???? (",)
      begin
      StringGrid1.Cells[0,j+l+1]:=IntToStr(j);

//intensitas:=(sqr(D[i,1])+sqr(E[i,1]))*(sqr(sin((StrToInt(Edit1.Text))*Q
[i,1]/2)))/(sqr(sin(Q[i,1]/2)));
      intensitas:=
(StrToInt(Edit1.Text))*(sqr(D[i,1])+sqr(C[i,1]))*(sqr(cos(Q[i,1]/2)));
      F[i,1]:=intensitas;
      StringGrid1.Cells[i+k+1,j+l+1]:=FloatToStr(F[i,1]);
      //Memo1.Lines.Add(IntToStr(i)+'  '+FloatToStr(F[i,1]));
      Series1.AddXY(i,F[i,1]);  //<<<---- Grafike mudhunn.... (_ _?)
      Chart3.Visible:=True;
      Series2.AddXY(i,F[i,1]);
      end;
    Memo1.Lines.Add(IntToStr(i)+'  '+FloatToStr(F[i,1]));
    pb:=i*4;
    ProgressBar1.Position:=4*i;
end;
  //menggambar spektrum
{JumlahBerkas:=StrToInt(Edit1.Text);
Layar:=JumlahBerkas*10;
  begin
  Spectrum:=TBitmap.Create;
  Spectrum.Height:=Image1.Height;
  Spectrum.Width:=((-Layar)*25-(Layar)*25);
  Spectrum.PixelFormat:=pf24bit;
  Spectrum.Canvas.Pen.Style:=psSolid;

  lebar:=0;
  {begin
    for i:=-k to k do
    begin
    application.ProcessMessages;
    Spectrum.Canvas.MoveTo(lebar,0);
```

```
    //WavelengthToRGB(PanjangGelombang,r,g,b);
    Spectrum.Canvas.Pen.Color:=RGB(round(r*F[i,j]), round(g*F[i,j]),
round(b*F[i,j]));
    Spectrum.Canvas.LineTo(lebar,Spectrum.Height-1);
    lebar:=lebar+1;
    end;}

    //Image1.Picture.Graphic:=Spectrum;
     end;

end;

procedure TForm1.Button5Click(Sender: TObject);
begin
Button1.Click;
Button2.Click;
Button3.Click;
Button4.Click;
ProgressBar1.Position:=pb;
end;

procedure TForm1.Button6Click(Sender: TObject);
var ft,fs:textfile;
  Strs : TStringList;
  i : integer;
begin
  if SaveDialog1.Execute then
    //create a TStringList to save the grid's contents to
    Strs := TStringList.Create;
    try
    //save the row and col counts
    Strs.Add(IntToStr(StringGrid1.RowCount));
    Strs.Add(IntToStr(StringGrid1.ColCount));

    //save each rows content as comma delimited text
    if StringGrid1.RowCount > 0 then begin
    for i := 0 to StringGrid1.RowCount -1 do
    Strs.Add( StringGrid1.Rows[i].CommaText );
    end;

    //save the stringlist to the file
    Strs.SaveToFile(SaveDialog1.FileName);
    finally
    Strs.Free;
    end;


end;

procedure TForm1.Button7Click(Sender: TObject);
var ft,fs:textfile;
  Strs : TStringList;
  i : integer;
begin
  if SaveDialog1.Execute then
    //create a TStringList to save the grid's contents to
    Strs := TStringList.Create;
    try
    //save the row and col counts
    Strs.Add(IntToStr(StringGrid2.RowCount));
    Strs.Add(IntToStr(StringGrid2.ColCount));

    //save each rows content as comma delimited text
```

```
    if StringGrid2.RowCount > 0 then begin
    for i := 0 to StringGrid2.RowCount -1 do
    Strs.Add( StringGrid2.Rows[i].CommaText );
    end;

    //save the stringlist to the file
    Strs.SaveToFile(SaveDialog1.FileName);
    finally
    Strs.Free;
    end;


end;

procedure TForm1.Button8Click(Sender: TObject);
var ft,fs:textfile;
  Strs : TStringList;
  i : integer;
begin
  if SaveDialog1.Execute then
    //create a TStringList to save the grid's contents to
    Strs := TStringList.Create;
    try
    //save the row and col counts
    Strs.Add(IntToStr(StringGrid3.RowCount));
    Strs.Add(IntToStr(StringGrid3.ColCount));

    //save each rows content as comma delimited text
    if StringGrid3.RowCount > 0 then begin
    for i := 0 to StringGrid3.RowCount -1 do
    Strs.Add( StringGrid3.Rows[i].CommaText );
    end;

    //save the stringlist to the file
    Strs.SaveToFile(SaveDialog1.FileName);
    finally
    Strs.Free;
    end;


end;

procedure TForm1.Button9Click(Sender: TObject);
begin
if SavePictureDialog1.Execute then
Chart1.SaveToBitmapFile(SavePictureDialog1.FileName);
end;

procedure TForm1.Button10Click(Sender: TObject);
begin
if SaveDialog1.Execute then
Memo1.Lines.SaveToFile(SaveDialog1.FileName);
end;

procedure TForm1.FormCreate(Sender: TObject);
begin
BMP_Celah:=TBitmap.Create;
   BMP_Celah.Width:=I_celah.Width;
   BMP_Celah.Height:=I_celah.Height;
   BMP_Layar:=TBitmap.Create;
   BMP_Layar.Width:=I_Layar.Width;
   BMP_Layar.Height:=I_Layar.Height;
```

```
   I_Layar.Picture.Assign(BMP_Layar);
   I_celah.Picture.Assign(BMP_Celah);
   jumlahTitik:=0;
    Memo2.Lines.Clear;
    Memo1.Lines.Clear;
end;

procedure TForm1.FormClose(Sender: TObject; var Action: TCloseAction);
begin
  BMP_Celah.Free;
  BMP_Layar.Free;

end;

procedure TForm1.I_celahMouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
   inc(jumlahTitik);
   titik[jumlahTitik]:=Point(X,Y);
   if jumlahTitik=1 then
   BMP_Celah.Canvas.Ellipse(x-2,y-2,x+2,y+2);
   if jumlahTitik=2 then
   begin
      BMP_Celah.Free;
      BMP_Celah:=TBitmap.Create;
      BMP_Celah.Width:=I_celah.Width;
      BMP_Celah.Height:=I_celah.Height;
      BMP_Celah.Canvas.MoveTo(titik[1].X,titik[1].Y);
   end;
   if jumlahTitik>1 then BMP_Celah.Canvas.LineTo(X,Y);
   BMP_Celah.Canvas.MoveTo(X,Y);
   I_celah.Picture.Assign(BMP_Celah);

end;

procedure TForm1.RadioButton1Click(Sender: TObject);

var
X1,X2,Y1,Y2,i,j,d,pusatx,pusaty,X,Y:integer;
jarakdatar:double;
begin
      I_celah.Picture:=nil;
      BMP_Celah.Free;
      BMP_Celah:=TBitmap.Create;
      BMP_Celah.Width:=I_celah.Width;
      BMP_Celah.Height:=I_celah.Height;
      //BMP_Celah.Canvas.MoveTo(titik[1].X,titik[1].Y);
//RadioButton1.Enabled:=false;
X1:=50;
Y1:=50;
X2:=70;
Y2:=70;
//d:=40;
  //BMP_Celah.Canvas.Rectangle(X1,Y1,X2,Y2);
  for i:=X1 to X2 do
  for j:=Y1 to Y2 do
  begin
  BMP_Celah.Canvas.Pixels[i,j]:=clblack;
  //BMP_Celah.Canvas.LineTo(X,Y);
  //I_celah.Picture.Assign(BMP_Celah);
  //pusatx:=Round(I_celah.Width/2);
  //pusaty:=Round(I_celah.Height/2)
  end;
```

```
   I_celah.Picture.Assign(BMP_Celah);
  {
    if Image1.Canvas.Pixels[i,j]=clblack then
   jarakdatar:=8*sqrt(sqr(abs(pusatx-j)-abs(100-i))+sqr(abs(pusaty-j)-
abs(100-i)));
   ListBox1.Items.Add(FloatToStr(jarakdatar));
   }

end;


procedure TForm1.RadioButton2Click(Sender: TObject);
var
X1,X2,Y1,Y2,ii,jj,d,pusatx,pusaty,X,Y:integer;
jarakdatar:double;
begin
      I_celah.Picture:=nil;
      BMP_Celah.Free;
      BMP_Celah:=TBitmap.Create;
      BMP_Celah.Width:=I_celah.Width;
      BMP_Celah.Height:=I_celah.Height;
      //BMP_Celah.Canvas.MoveTo(titik[1].X,titik[1].Y);
//RadioButton1.Enabled:=false;
X1:=50;
Y1:=50;
X2:=70;
Y2:=70;
//d:=40;

  BMP_Celah.Canvas.Ellipse(X1,Y1,X2,Y2);

  for ii:=X1 to Y1 do
  for jj:=X2 to Y2 do
  begin
  BMP_Celah.Canvas.Pixels[ii,jj]:=clblack;

   I_celah.Picture.Assign(BMP_Celah);

end;

    end;
procedure TForm1.Button11Click(Sender: TObject);
var i,j,masuk:integer;
begin
    BMP_Celah.Canvas.LineTo(titik[1].X,titik[1].Y);
    I_celah.Picture:=nil;


    for j:=1 to I_celah.Height do
    begin
      masuk:=0;
      for i:=1 to I_celah.Width do
      begin

         if (masuk=1) and (BMP_Celah.Canvas.Pixels[i+1,j]=clblack) then
masuk:=2;;
         if (masuk=0) and (BMP_Celah.Canvas.Pixels[i,j]=clblack) and
(BMP_Celah.Canvas.Pixels[i-1,j]=clwhite) then masuk:=1;

         if masuk=1 then I_celah.Canvas.Pixels[i,j]:=clblack;
      end;
      if masuk<>2 then for i:=1 to I_celah.Width do
I_celah.Canvas.Pixels[i,j]:=clwhite;
```

```
    end;
    BMP_Celah.Assign(I_celah.Picture);
    I_celah.Picture:=nil;
    I_celah.Picture.Assign(BMP_Celah);

end;

procedure TForm1.Button12Click(Sender: TObject);
begin
     BMP_Celah.Free;
     BMP_Celah:=TBitmap.Create;
     BMP_Celah.Width:=I_celah.Width;
     BMP_Celah.Height:=I_celah.Height;
     I_celah.Picture.Assign(BMP_Celah);
     jumlahTitik:=0;

end;
 function urutan_warna(urutan:integer):Tcolor;
begin
    case urutan of
      1:Result:=clBlack;
      2:Result:=clBlack;
      3:Result:=clBlack;
      4:Result:=clBlack;
      5:Result:=clBlack;
      6:Result:=clBlack;
      7:Result:=clBlack;
      8:Result:=clBlack;
      {2:Result:=clGray;
      3:Result:=clRed;
      4:Result:=clMaroon;
      5:Result:=clLime;
      6:Result:=clNavy;
      7:Result:=clSkyBlue;
      8:Result:=clMoneyGreen;
      9:Result:=clSilver;
      10:Result:=clAqua;
      11:Result:=clFuchsia;}
      {12:Result:=clOlive;
      13:Result:=clBlue;
      14:Result:=clGreen;
      15:Result:=clYellow;}
      12:Result:=clWhite;
      13:Result:=clWhite;
      14:Result:=clWhite;
      15:Result:=clWhite;
      16:Result:=clWhite;
    end;
end;

procedure TForm1.Button13Click(Sender: TObject);
var i,j,m,n,pusatx,pusaty,pusatcelahx,pusatcelahy:integer;
    lamda,totalR,totalI,TOTAL,
    jarakdatar,jarak,nilai,nilaibesar,k,max:real;
begin
    I_Layar.Picture:=nil;
    pusatx:=Round(BMP_Layar.Width/2);
    pusaty:=Round(BMP_Layar.Height/2);
    pusatcelahx:=Round(BMP_Celah.Width/2);
    pusatcelahy:=Round(BMP_Celah.Height/2);
    lamda:=100;
    k:=2*pi/lamda;
    max:=1;
```

```
    //for n:=pusaty to BMP_Layar.Height do
    n:=pusaty;
    begin
       //Label1.Caption:=inttostr(n);
       Series4.Clear;
       for m:=1 to BMP_Layar.Width do
       begin
          totalR:=0;
          totalI:=0;
          ProgressBar2.Position:=m;
          //for j:=1 to BMP_Celah.Height do
          j:=pusatcelahy;
          begin
             for i:=1 to BMP_Celah.Width do
             if BMP_Celah.Canvas.Pixels[i,j]=clblack then
             begin

                    Application.ProcessMessages;
                    jarakdatar:=8*sqrt(sqr(abs(pusatx-m)-abs(pusatcelahx-
i))+sqr(abs(pusaty-n)-abs(pusatcelahy-j)));

                    jarak:=sqrt(sqr(jarakdatar)+sqr(100));

                    totalR:=totalR+sqr(1*cos(k*jarak));
                    //totalI:=totalI+sqr(1*sin(k*jarak));
                   end;
          if (m=100) then
          TOTAL:=max
              else
             // begin
          TOTAL:=max*3*(totalR)*sqr(sin (jarak*k))/sqr(jarak*k);
          //if total>0 then
          begin
                //ShowMessage(floattostr(TOTAL));
                //SG_real.Cells[m,n]:=floattostr(TOTAL);

                Application.ProcessMessages;
                memo2.Lines.Add(floattostr(TOTAL));

BMP_Layar.Canvas.Pixels[m,n]:=urutan_warna(round(TOTAL/5));
                Series4.AddXY(m,TOTAL/5,'',clred);
                I_Layar.Picture.Assign(BMP_Layar);
                //ShowMessage(inttostr(round(TOTAL/4375)));
                //SG_Imaginer.Cells[m,n]:=floattostr(totalI);
          end;
       end;
    end;

end;
  end;

         //end;
procedure TForm1.Button14Click(Sender: TObject);
begin
if SaveDialog1.Execute then
Memo2.Lines.SaveToFile(SaveDialog1.FileName);
end;



end.
```