

LAMPIRAN

Tabel Data unit pembangkit thermal pada PT. PLN (Persero) Indonesia Power dan PT PJB

No.	Nama Pembangkit	Operasi	Cap (MW)	Cap Factor	Max MW	Min MW	Thermal Heat Rate kCal/kWh	Ramp rate MW/min	Fuel	Fuel Cost
PT Indonesia Power										
1.	PLTU Suralaya 1	21-08-84	400	0.70	400	200	2.430	10.00	Coal	155.0
2.	PLTU Suralaya 2	10-06-85	400	0.70	400	200	2.414	10.00	Coal	155.0
3.	PLTU Suralaya 3	24-08-88	400	0.70	400	200	2.430	10.00	Coal	155.0
4.	PLTU Suralaya 4	24-04-89	400	0.70	400	200	2.415	10.00	Coal	155.0
5.	PLTU Suralaya 5	16-12-96	600	0.70	600	300	2.380	5.00	Coal	155.0
6.	PLTU Suralaya 6	26-03-97	600	0.70	600	300	2.387	5.00	Coal	155.0
7.	PLTU Suralaya 7	19-09-97	600	0.70	600	300	2.385	5.00	Coal	155.0
8.	PLTU Priok 3	18-10-72	50	0.00	50	25	2.813	5.00	MFO	400.0
9.	PLTU Priok 4	05-08-72	50	0.00	50	25	2.813	5.00	MFO	400.0
10.	PLTG Priok 4	23-05-77	48.8	0.00	48.8	24.4	2.813	0.00	Gas	2.0
11.	PLTG Priok 5	30-05-77	48.8	0.00	48.8	24.4	3.071	0.00	Gas	2.0
12.	PLTG Priok 6	30-05-77	48.8	0.00	48.8	24.4	2.785	0.00	Gas	2.0
13.	PLTG Priok 7	30-05-77	48.8	0.70	48.8	24.4	2.785	0.00	Gas	2.0
14.	PLTGU PriokGT1-1	20-10-93	130	0.70	130	65.0	2.877	6.50	Gas	2.45
14.	PLTGU PriokGT1-2	16-11-93	130	0.70	130	65.0	2.842	6.50	Gas	2.45
14.	PLTGU PriokGT1-3	21-12-93	130	0.70	130	65.0	2.877	6.50	Gas	2.45
14.	PLTGU PriokGT1-0	17-09-94	200	0.70	200	100	1.960	6.50	Gas	2.45
15.	PLTGU PriokGT2-1	15-01-94	130	0.70	130	65.0	2.872	6.50	Gas	2.45
15.	PLTGU PriokGT2-2	15-02-94	130	0.70	130	65.0	2.990	6.50	Gas	2.45
15.	PLTGU PriokGT2-3	15-01-94	130	0.70	130	65.0	3.012	6.50	Gas	2.45
15.	PLTGU PriokGT2-0	15-01-94	200	0.70	200	100	1.960	6.50	Gas	2.45
16.	PLTU TB Lorok 1	25-09-78	50	0.00	50	25	2.751	0.75	MFO	430.0
17.	PLTU TB Lorok 2	17-01-78	50	0.00	50	25	2.705	0.75	MFO	430.0
18.	PLTU TB Lorok 3	07-02-83	200	0.00	200	100	2.375	3.00	MFO	430.0
19.	PLTGU TBLork1-1	31-08-93	109.65	0.70	109.65	54.83	3.399	6.00	HSD	630.0
19.	PLTGU TBLork1-2	10-03-93	109.65	0.70	109.65	54.83	3.262	6.00	HSD	630.0
19.	PLTGU TBLork1-3	21-08-93	109.65	0.70	109.65	54.83	3.278	6.00	HSD	630.0
19.	PLTGU TBLork1-0	31-08-93	188	0.70	188	94	1.994	6.00	HSD	630.0
20.	PLTGU TBLork2-1	24-07-96	109.65	0.70	109.65	54.83	3.242	6.00	HSD	630.0
20.	PLTGU TBLork2-2	30-02-96	109.65	0.70	109.65	54.83	3.283	6.00	HSD	630.0
20.	PLTGU TBLork2-3	09-04-96	109.65	0.70	109.65	54.83	3.237	6.00	HSD	630.0
20.	PLTGU TBLork2-0	16-05-97	188	0.00	188	94	1.867	6.00	HSD	630.0
21.	PLTU Perak 3	00-00-78	50	0.00	50	25	2.766	1.00	MFO	400.0
22.	PLTU Perak 4	00-00-78	50	0.00	50	25	2.769	1.00	MFO	400.0
23.	PLTG Sunyaragi 1	06-06-76	20.03	0.00	20.03	10.02	4.731	0.00	HSD	611.2
24.	PLTG Sunyaragi 2	21-01-76	20.03	0.00	20.03	10.02	4.117	0.00	HSD	611.2
25.	PLTG Sunyaragi 3	26-11-76	20.10	0.00	20.10	10.02	3.997	0.00	HSD	611.2
26.	PLTG Sunyaragi 4	30-08-76	20.10	0.00	20.10	10.02	4.638	0.00	HSD	611.2
27.	PLTG Cilacap 1	26-08-96	29.00	0.00	29.00	14.50	4.991	0.00	HSD	611.2
28.	PLTG Cilacap 2	15-10-96	29.00	0.00	29.00	14.50	5.323	0.00	HSD	611.2
29.	PLTG Pesanggarn 1	16-02-85	21.35	0.00	21.35	10.68	5.872	2.00	HSD	617.2
30.	PLTG Pesanggarn 2	19-05-93	21.35	0.00	21.35	10.05	5.319	2.00	HSD	617.2
31.	PLTG Pesanggarn 3	07-07-94	42.00	0.00	42.00	21.00	4.625	1.00	HSD	617.2
32.	PLTG Pesanggarn 4	20-08-94	42.00	0.00	42.00	21.00	4.380	1.00	HSD	617.2
33.	PLTD Pesanggarn 1	00-00-89	75.85	0.00	75.85	21.00	2.640	0.00	HSD	617.2
34.	PLTG Gilimanuk 1	29-07-97	133.80	0.00	133.80	50.00	4.161	0.00	HSD	617.2
35.	PLTP Kamojang1-3	13-09-87	140.00	0.00	140.00	70.00	1.475	0.00	GT	89.6
36.	PLTP G-Salak 1-3	16-07-97	165.00	0.00	165.00	82.50	8.225	0.00	GT	0.05
37.	PLTP Darajat	10-06-94	55.00	0.00	55.00	37.50	8.400	0.00	GT	0.03
38.	PLTGU GratiGT1-1	28-09-96	100.75	0.70	100.75	50.38	3.168	6.70	HSD	617.2
38.	PLTGU GratiGT1-2	10-07-96	100.75	0.70	100.75	50.38	3.144	6.70	HSD	617.2
38.	PLTGU GratiGT1-3	10-02-96	100.75	0.70	100.75	50.38	3.230	6.70	HSD	617.2
38.	PLTGU GratiGT1-0	23-01-97	154.58	0.70	154.58	92.50	2.120	6.70	HSD	617.2
39.	PLTGU GratiGT2-1	24-07-96	100.75	0.70	100.75	50.38	3.168	6.70	HSD	617.2
39.	PLTGU GratiGT2-2	30-08-96	100.75	0.70	100.75	50.38	3.144	6.70	HSD	617.2
39.	PLTGU GratiGT2-3	09-04-96	100.75	0.70	100.75	50.38	3.230	6.70	HSD	617.2
39.	PLTGU GratiGT2-0	16-05-97	154.58	0.70	154.58	92.50	2.531	6.70	HSD	617.2

Sumber: Power Plant Data Sheet PT. PLN (persero) Indonesia Power

No.	Nama Pembangkit	Operasi	Cap (MW)	Cap Factor	Max MW	Min MW	Thermal Heat Rate kCal/kWh	Ramp rate MW/min	Fuel	Fuel Cost
PT PJB										
1.	PLTU Paiton 1		400	0.70	400	200	2.436	10.00	Coal	71.06
2.	PLTU Paiton 2		400	0.70	400	200	2.436	10.00	Coal	71.06
3.	PLTU M.Karang 1		100	0.70	100	50	2.540	5.00	MFO	259.3
4.	PLTU M.Karang 2		100	0.70	100	50	2.540	5.00	MFO	258.3
5.	PLTU M.Karang 3		100	0.70	100	50	2.540	5.00	MFO	258.3
6.	PLTU M.Karang 4		200	0.00	200	100	2.375	5.00	Gas	2.45
7.	PLTU M.Karang 5		200	0.00	200	100	2.375	5.00	Gas	2.45
8.	PLTGU M.Kr.GT1-1		108	0.70	100	54	3.168	6.50	Gas	2.45
8.	PLTGU M.Kr.GT1-2		108	0.70	100	54	3.144	6.50	Gas	2.45
8.	PLTGU M.Kr.GT1-3		108	0.70	100	54	3.230	6.50	Gas	2.45
8.	PLTGU M.Kr.GT1-0		185	0.70	170	92.50	2.121	6.50	Gas	2.45
9.	PLTU Gresik 1		100	0.70	100	50	2.540	0.75	MFO	259.3
10.	PLTU Gresik 2		100	0.70	100	50	2.540	0.75	MFO	259.3
11.	PLTU Gresik 3		200	0.70	200	100	2.375	0.75	Gas	2.53
12.	PLTU Gresik 4		200	0.70	200	100	2.375	0.75	Gas	2.53
13.	PLTG Gresik 1		20	0.00	20	10	4.640	0.75	MFO	259.3
14.	PLTG Gresik 2		20	0.00	20	10	4.640	0.75	MFO	259.3
15.	PLTG Gresik 3		20	0.00	20	10	4.640	0.75	MFO	259.3
16.	PLTG Gresik 4		20	0.00	20	10	4.640	0.75	MFO	259.3
17.	PLTG Gresik 5		20	0.00	20	10	4.640	0.75	MFO	259.3
18.	PLTGU GresikGT1-1		112	0.70	112	56	3.168	6.00	HSD	650
18.	PLTGU GresikGT1-2		112	0.70	112	56	3.144	6.00	HSD	650
18.	PLTGU GresikGT1-3		112	0.70	112	56	3.230	6.00	HSD	650
18.	PLTGU GresikGT1-0		189	0.70	189	94.50	2.120	6.00	HSD	650
19.	PLTGU GresikGT2-1		112	0.70	112	56	3.168	6.00	HSD	650
19.	PLTGU GresikGT2-2		112	0.70	112	56	3.144	6.00	HSD	650
19.	PLTGU GresikGT2-3		112	0.70	112	56	3.230	6.00	HSD	650
19.	PLTGU GresikGT2-0		189	0.70	189	94.50	2.120	6.00	HSD	650
20.	PLTGU GresikGT3-1		112	0.70	112	56	3.168	6.00	Gas	2.53
20.	PLTGU GresikGT3-2		112	0.70	112	56	3.144	6.00	Gas	2.53
20.	PLTGU GresikGT3-3		112	0.70	112	56	3.230	6.00	Gas	2.53
20.	PLTGU GresikGT3-0		189	0.70	189	94.50	2.120	6.00	Gas	2.53
21.	PLTGU M.Tw.GT1-1		145	0.70	145	72	4.161	6.00	MFO	259.3
21.	PLTGU M.Tw.GT1-2		145	0.70	145	72	4.161	6.00	MFO	259.3
21.	PLTGU M.Tw.GT1-3		145	0.70	145	72	4.161	6.00	MFO	259.3
21.	PLTGU M.Tw.GT1-0		200	0.70	200	100	2.773	6.00	MFO	259.3

Sumber: Power Plant Data Sheet PT. PLN (persero) PJB

No.	No. Unit	A	B	C	MaxCap (MW)	MinCap (MW)	MUT (h)	MDT (h)	Biaya tanpa beban (\$/h)	Status awal	Harga bahan bakar (\$/MCal)	Biaya start up (\$/kWh)
PT Indonesia Power												
1.	Unit 1	388143.98	1306.00	6.18100	381.00	240.00	24	24	12.08	24	0.004259	20.13
2.	Unit 2	388143.98	1306.00	6.18100	381.00	240.00	24	24	12.08	24	0.004259	20.13
3.	Unit 3	388143.98	1306.00	6.18100	381.00	240.00	24	24	12.08	24	0.004259	20.13
4.	Unit 4	388143.98	1306.00	6.18100	381.00	240.00	24	24	12.08	24	0.004259	20.13
5.	Unit 5	561.00	7.92	0.00156	579.00	340.00	48	48	18.12	24	0.004259	30.20
6.	Unit 6	561.00	7.92	0.00156	579.00	340.00	48	48	18.12	24	0.004259	30.20
7.	Unit 7	737.00	10.98	2.10100	508.00	150.00	24	24	2.09	24	0.009616	3.48
8.	Unit 8	737.00	10.98	2.10100	508.00	150.00	24	24	3.03	24	0.009616	5.04
9.	Unit 9	660.80	25.92	0.00413	55.00	30.00	2	2	0.00	24	0.000005	0.00
10.	Unit 10	660.80	25.92	0.00413	55.00	30.00	2	2	0.00	24	0.000005	0.00
11.	Unit 11	660.80	25.92	0.00413	55.00	30.00	2	2	0.00	24	0.000005	0.00
12.	Unit 12	650.70	28.12	0.00209	50.00	20.00	2	2	0.00	24	0.008960	0.00
13.	Unit 13	650.70	28.12	0.00209	50.00	20.00	2	2	0.00	24	0.008960	0.00
14.	Unit 14	650.70	28.12	0.00209	50.00	20.00	2	2	0.00	24	0.008960	0.00
15.	Unit 15	451.01	22.10	0.00200	55.00	37.50	2	2	0.00	24	0.000003	0.00
16.	Unit 16	260.00	20.00	0.00980	50.00	25.00	24	12	2.79	24	0.009616	4.65
17.	Unit 17	260.00	20.00	0.00980	50.00	25.00	24	12	2.79	24	0.009616	4.65
18.	Unit 18	78.00	7.97	0.00480	200.00	100.00	24	12	4.55	24	0.009616	7.68
19.	Unit 19	696.03	877.00	1.92000	400.00	150.00	24	12	2.45	24	0.005771	1.36
20.	Unit 20	696.25	877.00	1.92000	400.00	150.00	24	12	2.45	24	0.005771	4.05
PT PJB												
1.	Unit 1	251800.0	2600	4.8000	508.00	150.00	4	4	3.97	24	0.009616	6.61
2.	Unit 2	375800.0	1968	9.7620	100.00	40.00	10	10	1.14	24	0.009616	1.90
3.	Unit 3	375800.0	1968	9.7620	100.00	40.00	10	10	1.14	24	0.009616	1.90
4.	Unit 4	303100.0	9.26	4.4000	100.00	70.00	10	10	4.55	24	0.009616	7.68
5.	Unit 5	303100.0	9.26	4.4000	100.00	70.00	10	10	4.55	24	0.009616	7.68
6.	Unit 6	388144.17	1306.15	6.1800	400.00	150.00	48	24	12.00	24	0.009616	20.0
7.	Unit 7	6000.00	2136.00	5.2800	100.00	40.00	48	48	1.14	24	0.005771	1.67
8.	Unit 8	6000.00	2136.00	5.2800	100.00	40.00	48	48	1.14	24	0.005771	1.67
9.	Unit 9	78.00	7.60	0.00200	200.00	50.00	48	48	4.54	24	0.009636	7.58
10.	Unit 10	914.3642	5.44	1.13780	550.00	250.00	48	0.5	1.97	24	0.009636	3.28
11.	Unit 11	998.8195	4.70	0.00240	550.00	250.00	48	0.5	1.97	24	0.009636	3.28
12.	Unit 12	955.8401	4.70	0.00201	550.00	250.00	48	0.5	1.97	24	0.009636	3.28

```

unit Unit1;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls, jpeg, ExtCtrls, pngimage;
type
  TForm1 = class(TForm)
    Button1: TButton;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Label4: TLabel;
    Image2: TImage;
    Image1: TImage;
    Label5: TLabel;
    procedure Button1Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;
var
  Form1: TForm1;
implementation

uses Unit2;
{$R *.dfm}
procedure TForm1.Button1Click(Sender: TObject);
begin
  FrmGenerator.show;
end;
end.

```

```

unit Unit2;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, Grids, StdCtrls, ComCtrls, MATH, inifiles;
type
  TFrmGenerator = class(TForm)
    BtnHasil: TButton;
    Label3: TLabel;
    Editjumlahunit: TEdit;
    PageControl1: TPageControl;
  end;

```

```

TabSheet1: TTabSheet;
TabSheet2: TTabSheet;
StrgKombinasi: TStringGrid;
BtnKombinasi: TButton;
StrgHasil: TStringGrid;
Btnisiunit: TButton;
TabSheet3: TTabSheet;
TabSheet4: TTabSheet;
StrgDataGen: TStringGrid;
StrgBeban: TStringGrid;
Edbanyakbeban: TEdit;
Label1: TLabel;
TabSheet5: TTabSheet;
Editprioritas: TEdit;
Button4: TButton;
Edit2: TEdit;
TabSheet7: TTabSheet;
StringGridinputoutput: TStringGrid;
btnLoad: TButton;
PBproses: TProgressBar;
TabSheet8: TTabSheet;
StrgUpDowntime: TStringGrid;
TabSheet9: TTabSheet;
StrgKombinasiUpDowntime: TStringGrid;
TabSheet10: TTabSheet;
StrgHasilAkhir: TStringGrid;
TabSheet11: TTabSheet;
StrgPalingMurah: TStringGrid;
Label2: TLabel;
Label4: TLabel;
OpenDialog1: TOpenDialog;
SaveDialog1: TSaveDialog;
pcdata: TPageControl;
btnSave: TButton;
procedure BtnHasilClick(Sender: TObject);
procedure BtnKombinasiClick(Sender: TObject);
procedure BtnisiunitClick(Sender: TObject);
procedure Button4Click(Sender: TObject);
procedure FormShow(Sender: TObject);
procedure btnLoadClick(Sender: TObject);
procedure btnSaveClick(Sender: TObject);
private
  { Private declarations }
public

```

```

    { Public declarations }
    function DecToBinStr(N, jumlahunit: Integer): string;
    function biyatotal(orde1,orde2,orde3,Fcost,daya: real): real;
    function hitungP(lambda,ord2,ord3:real):real;
    function hitunglambdabaruu(lambda, err: real; jml: integer): real;
end;
var
    FrmGenerator: TFrmGenerator;
    min:array[1..550000] of real;
    max:array[1..550000] of real;
    hargaakhir:array[1..550000] of real;
    beban:array[1..550000] of real;
    biyastart:array[1..550000] of real;
    biyafuel:array[1..550000] of real;
    waktudowntime:array[1..550000] of integer;
    waktuuptime:array[1..550000] of integer;
    statustime:array[1..550000] of integer;
    unitprioritas,hitungprioritas:array[1..550000] of integer;
    orde1,orde2,orde3 : array [1..550000] of real;
    fungsibiayafuelorde1,fungsibiayafuelorde2,fungsibiayafuelorde3 : array [1..550000]
of real;
    hasil : array [1..550000] of real;
    lambdabaruu: real;
    outputP : array[1..550000] of real;
    ikutlambda : array[1..550000] of integer;
const
    lambda : real =6;
implementation
    {$R *.dfm}
    function TFrmGenerator.DecToBinStr(N, jumlahunit: Integer): string;
var
    S, depan: string;
    i: Integer;
begin
    //mengubah angka ke biner
    for i:=1 to SizeOf(N)*8 do
        begin
            if N<0 then
                S:=S+'1'
            else
                S:=S+'0';
            N:=N shl 1;
        end;
    Delete(S, 1, Pos('1', S)-1);

```

```

depan := StringOfChar('0', (jumlahunit - Length(S)));
Result:=depan + S;
end;
procedure TFrmGenerator.BtnisiunitClick(Sender: TObject);
var i,jumlahunit,banyakbeban : integer;
begin
//tampilan data generator stringgrid
jumlahunit:=StrToIntDef(Editjumlahunit.Text, 0);
StrgDataGen.cells[1,0]:='No. Unit';
StrgDataGen.cells[2,0]:='Min MW';
StrgDataGen.cells[3,0]:='Max MW';
StrgDataGen.cells[4,0]:='StartUp Cost';
StrgDataGen.cells[5,0]:='Fuel Cost';
StrgDataGen.cells[6,0]:='Min Up Time';
StrgDataGen.cells[7,0]:='Min Down Time';
StrgDataGen.cells[8,0]:='Status Pembangkit';
StrgDataGen.RowCount:= jumlahunit +1;

for i := 1 to jumlahunit do
begin
StrgDataGen.cells[1,i]:=inttostr(i);
end;
// tampilan karakteristik
stringgridinputoutput.cells[0,0]:='No. Unit';
stringgridinputoutput.cells[1,0]:='Orde 1';
stringgridinputoutput.cells[2,0]:='Orde 2';
stringgridinputoutput.cells[3,0]:='Orde 3';
stringgridinputoutput.RowCount:= jumlahunit +1 ;
stringgridinputoutput.ColCount:= 4;
for i := 1 to jumlahunit do
begin
stringgridinputoutput.cells[0,i]:=inttostr(i);
end;
//tampilan Pembebanan Stringgrid
banyakbeban:=strtointdef(Edbanyakbeban.Text, 0);
StrgBeban.cells[0,0]:='Periode';
StrgBeban.cells[1,0]:='Beban (W)';
StrgBeban.ColCount:=2;
StrgBeban.RowCount:=banyakbeban +1;
for i := 1 to banyakbeban+1 do
begin
StrgBeban.cells[0,i]:='Periode ' + inttostr(i);
end;
end;
end;

```

```

procedure TFrmGenerator.BtnKombinasiClick(Sender: TObject);
var i,j: integer;
    jumlahunit: integer;
    banyakkombinasi,banyakbeban: integer;
    mintotal,maxtotal,start,fuel,maxuptime,mindowntime:real;
    hasilbiner : string;
    jumlaaha,jumlahb,jumlahax:real;
    a,b:array [1..10] of real;
    lambda : real;
    posisi: integer;
    banyakprioritas,jumlahbeban,jumlahkombinasi,hitung: integer;
    k,sementaraprio : integer;
    sementara:real;
    jumlahprioritas,jumlahnonprioritas:integer;
    sisabeban:real;
begin
//INISIALISASI jumlah unit dan banyak beban dari edit text
    jumlahunit:=StrToIntDef(Editjumlahunit.Text,0);
    banyakbeban:=strtointdef(Edbanyakbeban.Text,0);
//INISIALISASI variabel banyak beban
    for i := 1 to banyakbeban do
        begin
            beban[i]:=StrToFloatDef(StrgBeban.cells[1,i],0);
        end;

//INISIALISASI VARIABEL Orde
for i := 1 to jumlahunit do
    begin
        orde1[i]:=StrToFloatDef(stringgridinputoutput.cells[1,i],0);
        orde2[i]:=StrToFloatDef(stringgridinputoutput.cells[2,i],0);
        orde3[i]:=StrToFloatDef(stringgridinputoutput.cells[3,i],0);
    end;
//INISIALISASI VARIABEL data generator
for i := 1 to jumlahunit do
    begin
        min[i]:=StrToFloatDef(StrgDataGen.cells[2,i],0);
        max[i]:=StrToFloatDef(StrgDataGen.cells[3,i],0);
        biayastart[i]:=strtofloatdef(StrgDataGen.Cells[4,i],0);
        biayafuel[i]:= strtofloatdef(StrgDataGen.Cells[5,i],0);
        waktuuptime[i]:=strtointdef(StrgDataGen.cells[6,i],0);
        waktudowntime[i]:=StrTointDef(StrgDataGen.Cells[7,i],0);
        statutime[i]:=StrTointDef(StrgDataGen.Cells[8,i],0);
        unitprioritas[i]:= i;
    end;

```



```

//tampilan kombinasi
StrgKombinasi.ColCount:= jumlahunit+banyakbeban+3;
StrgKombinasi.cells[0,0]:='nomor unit';
for i := 1 to jumlahunit do
begin
  StrgKombinasi.Cells [i,0]:='unit ' + inttostr(i)
end;
StrgKombinasi.cells[jumlahunit+1,0]:='mintotal';
StrgKombinasi.cells[jumlahunit+2,0]:='maxtotal';
for i := jumlahunit+3 to jumlahunit+2+banyakbeban do
begin
  StrgKombinasi.Cells [i,0]:='Periode ' + inttostr(i-(jumlahunit+2))
end;
////////////////////////////////////
//distribusi biner ke masing2 kotak
banyakkombinasi:=trunc((power(2,jumlahunit))-1);
StrgKombinasi.RowCount:=banyakkombinasi+1;

for i := 1 to banyakkombinasi do
begin
  StrgKombinasi.cells[0,i]:= inttostr(i);
  hasilbiner:=DecToBinStr(i,jumlahunit);
  mintotal:=0;
  maxtotal:=0;
  for j := 1 to jumlahunit do
  begin
    StrgKombinasi.cells[j,i]:= hasilbiner[j];
    if hasilbiner[j] = '1' then
    begin
      mintotal := mintotal + min[j];
      maxtotal := maxtotal + max[j];
    end;
  end;
  StrgKombinasi.cells[jumlahunit+1,i]:=FloatToStr(mintotal);
  StrgKombinasi.cells[jumlahunit+2,i]:=FloatToStr(maxtotal);
  for j := jumlahunit+3 to jumlahunit+2+banyakbeban do
  begin
    if (beban[j-(jumlahunit+2)] >= mintotal) and (beban[j-(jumlahunit+2)] <=
maxtotal) then
    begin
      StrgKombinasi.Cells[j,i]:='1';
    end else
    begin
      StrgKombinasi.Cells[j,i]:='0';
    end;
  end;
end;

```

```

    end;
end;
end;
for i := 1 to jumlahunit do
begin
    fungsibiayafuelorde1[i]:= orde1[i]*biayafuel[i];
    fungsibiayafuelorde2[i]:= orde2[i]*biayafuel[i];
    fungsibiayafuelorde3[i]:= orde3[i]*biayafuel[i];
end;
////////////////////////////////////
//isi up/downtime

StrgUpDowntime.RowCount:=banyakbeban+2;
StrgUpDowntime.ColCount:=jumlahunit+1;
for i:= 0 to banyakbeban do
begin
    StrgUpDowntime.cells[0,i+1]:='Periode '+IntToStr(i);
end;
for i := 1 to jumlahunit do
begin
    StrgUpDowntime.cells[i,0]:='unit '+IntToStr(i);
    posisi:=statustime[i];
    if posisi > 0 then
    begin
        StrgUpDowntime.cells[i,1]:='1';
    end else
    begin
        StrgUpDowntime.cells[i,1]:='0';
    end;
    for j := 1 to banyakbeban do
    begin
        if posisi>0 then
        begin
            if posisi = waktuuptime[i] then
            begin
                posisi :=-1;
            end else
            begin
                posisi:= posisi+1;
            end;
        end else
        begin
            if abs(posisi) = waktudowntime[i] then
            begin

```

```

        posisi :=1;
    end else
    begin
        posisi:= posisi - 1;
    end;
end;
if posisi > 0 then
begin
    StrgUpDowntime.cells[i,j+1]:='1';
end else
begin
    StrgUpDowntime.cells[i,j+1]:='0';
end;
end;
end;
////////////////////////////////////
//isi kombinasi up/downtime
StrgKombinasiUpDowntime.RowCount:=1;
StrgKombinasiUpDowntime.ColCount:=jumlahunit+4;
//hitung banyak prioritas
jumlahunit:=StrToIntDef(editjumlahunit.Text,0);
if jumlahunit mod 2 = 0 then
begin
    banyakprioritas:=(jumlahunit div 2);
end else
begin
    banyakprioritas:=((jumlahunit div 2)+1);
end;
jumlahbeban:=StrToIntDef(Edbanyakbeban.Text,0);
jumlahkombinasi:= StrgKombinasi.RowCount;
hitung:=0;
StrgKombinasiUpDowntime.Cells[0,0]:= 'State ';
StrgKombinasiUpDowntime.Cells[1,0]:= 'Beban ';
//menentukan unit prioritas
for i := 1 to (jumlahunit -1) do
begin
    for j := (i+1) to jumlahunit do
    begin
        if biayafuel[i] > biayafuel[j] then
        begin
            sementara:= biayafuel[i];
            biayafuel[i] := biayafuel[j];
            biayafuel[j] := sementara;
            sementaraprio := unitprioritas[i];

```

```

    unitprioritas[i] := unitprioritas[j];
    unitprioritas[j] := sementaraprio;
end;
end;
end;
//isi header
for k := 1 to jumlahunit do
begin
    StrgKombinasiUpDowntime.Cells[k+1,0]:= 'P'+IntToStr(unitprioritas[k]);
end;
StrgKombinasiUpDowntime.Cells[jumlahunit+2,0]:='Memenuhi';
StrgKombinasiUpDowntime.Cells[jumlahunit+3,0]:='Periode';
//looping tabel hasil
for j := 1 to jumlahbeban do
begin
    for i := 1 to jumlahkombinasi-1 do
    begin
        if StrgKombinasi.Cells[jumlahunit+2+j,i] = '1' then
        begin
            //isi kombinasi 1 tabel hasil
            hitung:=hitung+1;
            StrgKombinasiUpDowntime.RowCount:=hitung + 1;
            StrgKombinasiUpDowntime.Cells[0,hitung] := 'Periode ' + IntToStr(j) + '(No '
+ IntToStr(i) + ')';
            StrgKombinasiUpDowntime.Cells[jumlahunit + 3,hitung] := IntToStr(j);
            //isi beban tabel hasil
            StrgKombinasiUpDowntime.Cells[1,hitung]:= FloatToStr(beban[j]);
            //hitungprioritas
            jumlahprioritas:=0;
            jumlahnonprioritas:=0;
            for k := 1 to jumlahunit do
            begin
                if StrgKombinasi.Cells[unitprioritas[k],i]='1' then
                begin
                    if k <= banyakprioritas then
                    begin
                        jumlahprioritas:=jumlahprioritas+1;
                        hitungprioritas[jumlahprioritas]:=k;
                    end else
                    begin
                        jumlahnonprioritas:=jumlahnonprioritas+1;
                    end;
                end;
            end;
        end;
    end;
end;
end;
end;

```

```

//isi kolom P(daya) unit
sisabeban:=beban[j];
mintotal:=0;
maxtotal:=0;
for k := 1 to jumlahunit do
begin
if StrgKombinasi.Cells[unitprioritas[k],i]='1' then
begin
if StrgUpDowntime.Cells[unitprioritas[k],j+1] = '0' then begin
StrgKombinasiUpDowntime.Cells[k+1,hitung]:= 'X';
end else
begin
StrgKombinasiUpDowntime.Cells[k+1,hitung]:= '1';
mintotal:=mintotal + min[unitprioritas[k]];
maxtotal:=maxtotal + max[unitprioritas[k]];
end;
end else
begin
StrgKombinasiUpDowntime.Cells[k+1,hitung]:= '0';
end;
end;
if (sisabeban >= mintotal) and (sisabeban <= maxtotal) then
begin
StrgKombinasiUpDowntime.Cells[jumlahunit+2,hitung]:= '1';
end else
begin
StrgKombinasiUpDowntime.Cells[jumlahunit+2,hitung]:= '0';
end;
end;
end;
end;
end;
procedure TFrmGenerator.BtnHasilClick(Sender: TObject);
var
jumlahkombinasi: integer;
i, j, k, l, m: integer;
jumlahunit, jumlahbeban, hitung, sementaraprio: integer;
sementara, sisabeban, sisabebanprio, totalP, er, hasilP, totalmaxprio: real;
jumlahprioritas, jumlahnonprioritas: integer;
banyakprioritas : integer;
hitungulang: boolean;
biayatotal, hitungbiaya, nilaiorde1, nilaiorde2: real;
nilaiorde3, nilaifcost, nilaidaya: real;
isidaya: string;

```

```

isijam, isijamlama, kombinasimurah: integer;
totalstartcost, totalcost, grandtotalcost, hargasementara: real;
begin
PBproses.Position:=0;
//hitung banyak prioritas
jumlahunit:=StrToIntDef(editjumlahunit.Text,0);
if jumlahunit mod 2 = 0 then
begin
    banyakprioritas:=(jumlahunit div 2);
end else
begin
    banyakprioritas:=((jumlahunit div 2)+1);
end;
StrgHasil.RowCount:=1;
StrgHasil.ColCount:=jumlahunit + 4;
jumlahbeban:=StrToIntDef(Edbanyakbeban.Text,0);
jumlahkombinasi:= StrgKombinasiUpDowntime.RowCount;
hitung:=0;
StrgHasil.Cells[0,0]:= 'State ';
StrgHasil.Cells[1,0]:= 'Beban ';
//menentukan unit prioritas
for i := 1 to (jumlahunit -1) do
begin
    for j := (i+1) to jumlahunit do
    begin
        if biayafuel[i] > biayafuel[j] then
        begin
            sementara:= biayafuel[i];
            biayafuel[i] := biayafuel[j];
            biayafuel[j] := sementara;
            sementaraprio := unitprioritas[i];
            unitprioritas[i] := unitprioritas[j];
            unitprioritas[j] := sementaraprio;
        end;
    end;
end;
//isi header
for k := 1 to jumlahunit do
begin
    StrgHasil.Cells[k+1,0]:= 'P'+IntToStr(unitprioritas[k]);
end;
StrgHasil.Cells[jumlahunit + 2, 0]:= 'Biaya Total';
StrgHasil.Cells[jumlahunit + 3, 0]:= 'Periode';
//looping tabel hasil

```

```

//for j := 1 to jumlahbeban do
//begin
PBproses.Max:=jumlahkombinasi;
for i := 1 to jumlahkombinasi-1 do
begin
  Application.ProcessMessages;
  PBproses.Position:=i;
  if StrgKombinasiUpDowntime.Cells[jumlahunit + 2, i] ='1' then
  begin
    //isi kombinasi 1 tabel hasil
    hitung:=hitung+1;
    StrgHasil.RowCount:=hitung + 1;
    StrgHasil.Cells[0,hitung] := StrgKombinasiUpDowntime.Cells[0, i];
    //isi beban tabel hasil
    StrgHasil.Cells[1,hitung]:= StrgKombinasiUpDowntime.Cells[1, i];
    //hitungprioritas
    jumlahprioritas:=0;
    jumlahnonprioritas:=0;
    totalmaxprio:=0;
    biayatotal:=0;
    for k := 1 to jumlahunit do
    begin
      if StrgKombinasiUpDowntime.Cells[k + 1, i]='1' then
      begin
        if k <= banyakprioritas then
        begin
          jumlahprioritas:=jumlahprioritas+1;
          totalmaxprio := totalmaxprio+max[unitprioritas[k]];
          hitungprioritas[jumlahprioritas]:=k;
        end else
        begin
          jumlahnonprioritas:=jumlahnonprioritas+1;
        end;
      end;
    end;
    //isi kolom P(daya) unit
    sisabeban:=StrToFloatDef(StrgKombinasiUpDowntime.Cells[1,i],0);
    //looping part 1 : bagi nilai min ke semua unit dengan status 1
    for k := 1 to jumlahunit do
    begin
      if StrgKombinasiUpDowntime.Cells[k+1,i]='1' then
      begin
        //kemungkinan prio <=1, non prio >=1

```

```

if (jumlahprioritas <=1) and (jumlahnonprioritas >= 1) then
begin
  StrgHasil.Cells[k+1,hitung]:= FloatToStr(min[unitprioritas[k]]);
  sisabeban:=sisabeban-min[unitprioritas[k]];
end;
//kemungkinan prio = 1 non = 0
if (jumlahprioritas =1) and (jumlahnonprioritas =0) then
begin
  StrgHasil.Cells[k+1,hitung]:= FloatToStr(sisabeban);
  sisabeban:=0;
end;
//kemungkinan prio lebih dari 1 dan non prio 0
if (jumlahprioritas >1) and (jumlahnonprioritas = 0) then
begin
  StrgHasil.Cells[k+1,hitung]:= '?';
end;
//kemungkinan prio lebih dari 1 dan non prio lebih dari 1
if (jumlahprioritas >1) and (jumlahnonprioritas >=1) then
begin
  if k <= banyakprioritas then
  begin
    StrgHasil.Cells[k+1,hitung]:= '?';
  end else
  begin
    StrgHasil.Cells[k+1,hitung]:= FloatToStr(min[unitprioritas[k]]);
    sisabeban:=sisabeban-min[unitprioritas[k]];
  end;
end;
end
else begin
  // status 0 / X
  StrgHasil.Cells[k+1,hitung]:= '0';
end;
end;
//looping part 2 : jika masih ada lebih seteah dibagi min
//      lalu diulang iterasi lambda
if sisabeban >0 then
begin
  sisabebanprio:=sisabeban;
  for k := 1 to jumlahunit do
  begin
    if (StrgKombinasiUpDowntime.Cells[k+1,i]='1') then
    begin
      //kemungkinan prio <=1, non prio >=1
    end
  end
end

```



```

if (jumlahprioritas <=1) and (jumlahnonprioritas >= 1) then
begin
//ShowMessage(IntToStr(j)+'-'+ IntToStr(i) +' : '+FloatToStr(sisabeban));
//cek maksimal daya prio cukup
if min[unitprioritas[k]] + sisabeban <= max[unitprioritas[k]] then
begin
StrgHasil.Cells[k+1,hitung]:= FloatToStr(min[unitprioritas[k]] +
                                sisabeban);

sisabeban:=0;
//break;
end else
begin
//cek maksimal daya prio tidak cukup
StrgHasil.Cells[k+1,hitung]:= FloatToStr(max[unitprioritas[k]]);
sisabeban:=(min[unitprioritas[k]] + sisabeban) - max[unitprioritas[k]];
//if sisabeban = 0 then break;
end;
end;
//totalmax unit prio yang hidup <= sisabeban
// (stelah dibagi dgn minus non prio yg hidup)
if jumlahprioritas >1 then
begin
//tanpa lambda
if totalmaxprio <= sisabebanprio then
begin
if k <= banyakprioritas then begin
//cek maksimal daya prio cukup
if sisabeban <= max[unitprioritas[k]] then
begin
StrgHasil.Cells[k+1,hitung]:= FloatToStr(sisabeban);
sisabeban:=0;
end else
//cek maksimal daya prio tidak cukup
begin
StrgHasil.Cells[k+1,hitung]:= FloatToStr(max[unitprioritas[k]]);
sisabeban:=sisabeban - max[unitprioritas[k]];
end;
end else begin
//cek maksimal daya non prio tidak cukup
if min[unitprioritas[k]] + sisabeban <= max[unitprioritas[k]] then
begin
StrgHasil.Cells[k+1,hitung]:= FloatToStr(min[unitprioritas[k]] +
                                sisabeban);

sisabeban:=0;

```

```

end else
//cek maksimal daya non prio tidak cukup
begin
  StrgHasil.Cells[k+1,hitung]:= FloatToStr(max[unitprioritas[k]]);
  sisabeban:=(min[unitprioritas[k]] + sisabeban) - max[unitprioritas[k]];
end;
end;
end else
begin
  // iterasi lambda
  totalP:=0;
  er:=1;
  for l := 1 to jumlahprioritas do
  begin
    // hitung Total P Awal
    ikutlambda[l]:=1;
    hasilP:=hitungP(lambda,
orde2[unitprioritas[hitungprioritas[l]],orde3[unitprioritas[hitungprioritas[l]]]);
    outputP[l]:=hasilP;
    totalP:=totalP + hasilP;
  end;
  //error tahap awal
  er:=sisabeban - totalP;
  while er >0.00001 do
  begin
    //itearasi selama er > 0.00001
    lambdabar:=hitunglambdabar(lambda,er,jumlahprioritas);
    totalP:=0;
    //Hitung Total P iterasi
    for l := 1 to jumlahprioritas do
    begin
      hasilP:=hitungP(lambdabar,
orde2[unitprioritas[hitungprioritas[l]],orde3[unitprioritas[hitungprioritas[l]]]);
      outputP[l]:=hasilP;
      totalP:=totalP + hasilP;
    end;
    er:=sisabeban - totalP;
  end;
  // jika hasil 1 unit P melebihi max
  if (jumlahprioritas>=2) then begin
    hitungulang := false;
    totalP:=0;
    for l := 1 to jumlahprioritas do
    begin

```

```

// cek apakah ada yang melebihi max
if (outputP[1] > max[unitprioritas[hitungprioritas[1]]]) then
begin
  //ShowMessage(FloatToStr(outputP[1])+
  floattostr(max[unitprioritas[hitungprioritas[1]]));
  outputP[1]:= max[unitprioritas[hitungprioritas[1]]];
  ikutlambda[1] := 0;
  hitungulang:=true;
end;
totalP:=totalP + outputP[1];
end;
//Jika ada yang melebihi, hitung lambda lagi
if hitungulang then
begin
  er:=sisabeban - totalP;
  //ShowMessage(IntToStr(j)+'-'+ IntToStr(i) +' : '+FloatToStr(er));
  while er >0.00001 do
  begin
    lambdabar:=hitunglambdabar(lambdabar,er,jumlahprioritas);
    //ShowMessage('lmbdbaru: '+FloatToStr(lambdabar));
    totalP:=0;
    for l := 1 to jumlahprioritas do
    begin
      if ikutlambda[l]=1 then
      begin
        //ShowMessage('1');
        hasilP:=hitungP(lambdabar,
orde2[unitprioritas[hitungprioritas[1]]],orde3[unitprioritas[hitungprioritas[1]]]);
        outputP[l]:=hasilP;
      end else
      begin
        //ShowMessage('0');
        hasilP:=outputP[l];
      end;
      totalP:=totalP + hasilP;
    end;
  end;
  er:=sisabeban - totalP;
end;
end;
//tampilkan hasil lambda
for l := 1 to jumlahprioritas do
begin
  StrgHasil.Cells[(hitungprioritas[l]+1),hitung]:= FloatToStr(outputP[l]);

```

```

        //StrgHasil.Cells[(hitungprioritas[l]+1),hitung]:= FloatToStr(er);
    end;
end;
end;
nilaiorde1:=orde1[unitprioritas[k]];
nilaiorde2:=orde2[unitprioritas[k]];
nilaiorde3:=orde3[unitprioritas[k]];
nilaifcost:=biayafuel[k];
nilaidaya:=StrToFloatDef(StrgHasil.Cells[(k+1), hitung], 0);
hitungbiaya:=biyatotal(nilaiorde1, nilaiorde2, nilaiorde3, nilaifcost,
nilaidaya);
biyatotal := biyatotal + hitungbiaya;
//if hitung = 1 then begin
    //ShowMessage(FloatToStr(nilaiorde1)+'-'+FloatToStr(nilaiorde2)+'-
'+FloatToStr(nilaiorde3)+'-'+FloatToStr(nilaifcost)+'-'+FloatToStr(nilaidaya)+'-
'+FloatToStr(hitungbiaya));
    //ShowMessage(StrgHasil.Cells[(k+1), hitung]);
    //end;
end else
begin
    StrgHasil.Cells[k+1,hitung]:= '0';
end;
end;
end;
StrgHasil.Cells[jumlahunit+2,hitung]:= FloatToStr(biyatotal);
StrgHasil.Cells[jumlahunit+3,hitung]:=
StrgKombinasiUpDowntime.Cells[jumlahunit+3,i];
end;
end;
////////////////////////////////////
//Start Cost
StrgHasilAkhir.RowCount:=StrgHasil.RowCount;
StrgHasilAkhir.ColCount:=jumlahunit + 2;
StrgHasilAkhir.Cells[0,0]:='State';
for i:=1 to jumlahunit do begin
    StrgHasilAkhir.Cells[i,0]:='Start Cost P'+IntToStr(unitprioritas[i]);
end;
StrgHasilAkhir.Cells[jumlahunit+1,0]:='Biaya + Start Cost';
//loop kombinasi
for i:=1 to StrgHasil.RowCount do begin
    //isi state
    StrgHasilAkhir.Cells[0,i]:=StrgHasil.Cells[0,i];
    totalstartcost:=0;
    for k:=1 to jumlahunit do begin

```

```

//isi cost
isijam:=StrToIntDef(StrgHasil.Cells[jumlahunit+3, i], 0);
if StrgUpDowntime.Cells[unitprioritas[k], isijam + 1] >
StrgUpDowntime.Cells[unitprioritas[k], isijam] then begin
  StrgHasilAkhir.Cells[k, i]:=FloatToStr(biayastart[unitprioritas[k]]);
  totalstartcost:= totalstartcost + biayastart[unitprioritas[k]];
end else begin
  StrgHasilAkhir.Cells[k, i]:='0';
end;
end;
//isi biaya akhir
totalcost:=StrToFloatDef(StrgHasil.Cells[jumlahunit + 2, i], 0);
grandtotalcost:=totalcost + totalstartcost;
StrgHasilAkhir.Cells[jumlahunit + 1, i]:=FloatToStr(grandtotalcost);
end;
////////////////////////////////////
//PALING MURAH
//label
StrgPalingMurah.ColCount:=jumlahunit + 2;
StrgPalingMurah.RowCount:=jumlahbeban + 1;
//isi header
for k := 1 to jumlahunit do
begin
  StrgPalingMurah.Cells[k,0]:= 'P'+IntToStr(unitprioritas[k]);
end;
StrgPalingMurah.Cells[jumlahunit + 1, 0]:= 'Biaya Total';

for j:=1 to jumlahbeban do begin
  hargasementara:=-1;
  for i:=1 to StrgHasil.RowCount do begin
    isijam:=StrToIntDef(StrgHasil.Cells[jumlahunit+3, i], 0);
    if isijam = j then begin
      if hargasementara = -1 then begin
        hargasementara:=StrToFloatDef(StrgHasilAkhir.Cells[jumlahunit + 1, i], 0);
        kombinasimurah:=i;
      end else begin
        if StrToFloatDef(StrgHasilAkhir.Cells[jumlahunit + 1, i], 0) < hargasementara
then begin
          hargasementara:=StrToFloatDef(StrgHasilAkhir.Cells[jumlahunit + 1, i], 0);
          kombinasimurah:=i;
        end;
      end;
    end;
  end;
end;
end;
end;

```

```

StrgPalingMurah.Cells[0,j]:=StrgHasilAkhir.Cells[0,kombinasimurah];
for k:=1 to jumlahunit do begin
  StrgPalingMurah.Cells[k,j]:=StrgHasil.Cells[k+1,kombinasimurah];
end;
StrgPalingMurah.Cells[jumlahunit+1,j]:=StrgHasilAkhir.Cells[jumlahunit +
1,kombinasimurah];
end;

{ sijamlama:=1;
hargaakhir[1]:=StrToFloatDef(StrgHasilAkhir.Cells[jumlahunit + 1, 1],0);
for i:= 1 to StrgHasilAkhir.RowCount do begin
  isijam:=StrToIntDef(StrgHasil.Cells[jumlahunit+3, i], 0);
  if hargaakhir[isijam] > StrToFloatDef(StrgHasilAkhir.Cells[jumlahunit + 1, i], 0)
then begin
  hargaakhir[isijam] := StrToFloatDef(StrgHasilAkhir.Cells[jumlahunit + 1, i], 0);
  kombinasimurah:=i;
end;
if isijam <> isijamlama then begin
  StrgPalingMurah.Cells[0, ]
end;
end;
}
//ISI TABEL PALING MURAH
//StrgPalingMurah.RowCount:=jumlahbeban + 1;
//StrgPalingMurah.ColCount:=2;
end;
procedure TFrmGenerator.Button4Click(Sender: TObject);
var jumlahprioritas : integer;
    jumlahunit,banyakbeban : integer;
    sementara: real;
    sementaraprio:integer;
    i,j:integer;
    jumlaa,jumlahb:real;
    a,b:array [1..10] of real;
    lambda : real;
begin
jumlahunit:=StrToIntDef(Editjumlahunit.Text,0);
banyakbeban:=strtointdef(Edbanyakbeban.Text,0);
if jumlahunit mod 2 = 0 then
begin
Editprioritas.Text:=IntToStr(jumlahunit div 2);
end
else
begin
Editprioritas.Text:=IntToStr((jumlahunit div 2)+1);

```

```

end;
jumlahprioritas:=strtoint(Editprioritas.Text);
for i := 1 to (jumlahunit -1) do
begin
  for j := (i+1) to jumlahunit do
  begin
    if biayafuel[i] > biayafuel[j] then
    begin
      sementara:= biayafuel[i];
      biayafuel[i] := biayafuel[j];
      biayafuel[j] := sementara;
      sementaraprio := unitprioritas[i];
      unitprioritas[i] := unitprioritas[j];
      unitprioritas[j] := sementaraprio;
    end;
  end;
end;
edit2.Text:="";
for i := 1 to jumlahprioritas do
begin
  ShowMessage(inttostr(unitprioritas[i]));
  Edit2.Text:=Edit2.Text+' ' + inttostr(unitprioritas[i]);
end;
{for j := 1 to banyakbeban do
begin
  for i := 1 to jumlahunit do
  begin
    a[i] := ((orde2[i]*biayafuel[i])/(2*biayafuel[i]*orde3[i]));
    b[i] := (1/(2*biayafuel[i]*orde3[i]));
    showmessage(floattostr(a[i]) + '-' + floattostr(b[i]))
  end;
  jumlaaha := 0;
  jumlahb := 0;
  for i := 1 to jumlahunit do
  begin
    jumlaaha := jumlaaha + a[i];
    jumlahb := jumlahb + b[i];
  end;

  showmessage('jumlaaha' + floattostr(jumlaaha));
  showmessage('jumlahb' + floattostr(jumlahb));
  lambda:= ((jumlaaha + beban[i])/jumlahb);
  showmessage('jumlah ' + floattostr(lambda))
end;}

```

```

end;
function TFrmGenerator.biyatotal(orde1, orde2, orde3, Fcost, daya: real): real;
begin
  Result:=(orde1*Fcost) + (orde2*Fcost*daya) + (orde3*Fcost*(daya*daya));
end;
procedure TFrmGenerator.FormShow(Sender: TObject);
begin
end;
procedure TFrmGenerator.btnLoadClick(Sender: TObject);
var
  data : TIniFile;
  jmlunit, jmlperiode, i: integer;
begin
  if OpenFileDialog1.Execute then
  begin
    if FileExists(OpenDialog1.FileName) then
    begin
      data := TIniFile.Create(OpenDialog1.FileName);
      Try
        //load jumlah unit dan beban
        Editjumlahunit.Text:= data.ReadString('Jumlah','unit', '0');
        jmlunit:=StrToIntDef(Editjumlahunit.Text,0);
        Edbanyakbeban.Text:= data.ReadString('Jumlah','periode', '0');
        jmlperiode:=StrToIntDef(Edbanyakbeban.Text,0);
        BtnisiunitClick(nil);
        //load datagenerator
        StrgDataGen.RowCount:=jmlunit + 1;
        for i:= 1 to jmlunit do
        begin
          StrgDataGen.Cells[1,i]:= IntToStr(i);
          StrgDataGen.Cells[2,i]:= data.ReadString('Unit'+inttostr(i),'min', '0');
          StrgDataGen.Cells[3,i]:= data.ReadString('Unit'+inttostr(i),'max', '0');
          StrgDataGen.Cells[4,i]:= data.ReadString('Unit'+inttostr(i),'startc', '0');
          StrgDataGen.Cells[5,i]:= data.ReadString('Unit'+inttostr(i),'fuelc', '0');
          StrgDataGen.Cells[6,i]:= data.ReadString('Unit'+inttostr(i),'minup', '0');
          StrgDataGen.Cells[7,i]:= data.ReadString('Unit'+inttostr(i),'mindown', '0');
          StrgDataGen.Cells[8,i]:= data.ReadString('Unit'+inttostr(i),'status', '0');
        end;
        //Load karakteristik input-output
        StringGridinputoutput.RowCount:=jmlunit+1;
        for i:= 1 to jmlunit do
        begin
          StringGridinputoutput.Cells[0,i]:= IntToStr(i);

```



```

StringGridinputoutput.Cells[1,i]:= data.ReadString('Unit'+inttostr(i),'orde1',
'0');
StringGridinputoutput.Cells[2,i]:= data.ReadString('Unit'+inttostr(i),'orde2',
'0');
StringGridinputoutput.Cells[3,i]:= data.ReadString('Unit'+inttostr(i),'orde3',
'0');
end;
//Loadbeban
StrgBeban.RowCount:=jmlperiode+1;
for i:= 1 to jmlperiode do
begin
StrgBeban.Cells[0,i]:= 'Periode '+ IntToStr(i);
StrgBeban.Cells[1,i]:= data.ReadString('Periode','periode'+IntToStr(i), '0');
end;

ShowMessage('Load Data Berhasil');
finally
data.Free;
end;
end;
end;
function TFrmGenerator.hitungP(lambda, ord2, ord3:real): real;
var
x: real;
begin
if ord3 <= 0 then
begin
result := 0;
exit;
end;
x := (lambda-ord2)/(2*ord3);
Result:=RoundTo(x,-6);
end;
function TFrmGenerator.hitunglambdabaruu(lambda, err: real; jml: integer): real;
var
dlam: real;
i,j : integer;
totalorde, hitungorde:real;
begin
totalorde:=0;
for i := 1 to jml do
begin
if ikutlambda[i] = 1 then

```

```

begin
  hitungorde:=1/(2*orde3[unitprioritas[hitungprioritas[i]]]);
end else
begin
  hitungorde:=0;
end;
//showmessage(IntToStr(ikutlambda[i]));
totalorde:=totalorde+hitungorde;
end;
dlam:= err / (totalorde);
result := lambda+dlam;
end;
procedure TFrmGenerator.btnSaveClick(Sender: TObject);
var
  data : TIniFile;
  jmlunit, jmlperiode, i: integer;
begin
  SaveDialog1.FileName:= 'UnitComData ' + FormatDateTime('yyyymmddhhnnss',
Now)+'.ini';
  if SaveDialog1.Execute then
  begin
    data:=TIniFile.Create(SaveDialog1.FileName);
    try
      //save Jumlah
      jmlunit:=StrToIntDef(Editjumlahunit.Text,0);
      jmlperiode:=StrToIntDef(Edbanyakbeban.Text,0);
      data.WriteInteger('Jumlah','Unit',jmlunit);
      data.WriteInteger('Jumlah','Periode',jmlperiode);
      //save data generator
      for i:= 1 to jmlunit do
      begin
        data.WriteString('Unit' + IntToStr(i),'min',StrgDataGen.Cells[2,i]);
        data.WriteString('Unit' + IntToStr(i),'max',StrgDataGen.Cells[3,i]);
        data.WriteString('Unit' + IntToStr(i),'startc',StrgDataGen.Cells[4,i]);
        data.WriteString('Unit' + IntToStr(i),'fuelc',StrgDataGen.Cells[5,i]);
        data.WriteString('Unit' + IntToStr(i),'minup',StrgDataGen.Cells[6,i]);
        data.WriteString('Unit' + IntToStr(i),'mindown',StrgDataGen.Cells[7,i]);
        data.WriteString('Unit' + IntToStr(i),'status',StrgDataGen.Cells[8,i]);
        //save data karakteristik input-output
        data.WriteString('Unit' + IntToStr(i),'orde1',StringGridinputoutput.Cells[1,i]);
        data.WriteString('Unit' + IntToStr(i),'orde2',StringGridinputoutput.Cells[2,i]);
        data.WriteString('Unit' + IntToStr(i),'orde3',StringGridinputoutput.Cells[3,i]);
      end;
    //SAVE data beban

```

```
for i:= 1 to jmlperiode do
begin
  data.WriteString('Periode' , 'periode' + IntToStr(i), StrgBeban.Cells[1,i]);
end;
  ShowMessage('Save Data Berhasil di ' + SaveDialog1.FileName);
finally
  data.Free;
end;
end;
end.
end.
```