

# LOLP (LOSS OF LOAD PROBABILITY)

Sekolah Tinggi Teknik - PLN

Agustus 2007

Yoga Prihastomo<sup>1</sup> Hari Suprianto<sup>2</sup>  
Saulus Silitonga<sup>3</sup>

---

## Pendahuluan

Dikarenakan saya tidak diperlihatkan makalah asli hasil seminar Bapak Saulus, maka dokumen ini adalah *hasil saduran dari internet*. Maklum saja, saya kuliah di STT PLN yang notabene adalah punya PT PLN, namun jurusan adalah Teknik Informatika. Suatu saat, saya diminta untuk mengkonversi program LOLP yang sudah dibuat dalam bahasa Pascal menjadi yang lebih visual. Dan karena saya bisanya Visual Basic 6, ya nanti akan saya tampilkan program yang sudah jadi dalam Pascal dan Visual Basic. Saya sendiri juga jujur tidak tau mengenai teori tentang LOLP ini, walau saya sudah buat programnya.

## Teori-Teori Pendahuluan

Perencanaan pembangkitan adalah suatu proses kegiatan perencanaan yang rumit yang bertujuan untuk mencari dan memilih suatu rencana yang optimal diantara beberapa alternatif rencana yang tersedia. Rencana yang optimal tersebut harus memenuhi total biaya pembangkitan yang minimum dengan batasan keandalan, lingkungan dan ketersediaan pendanaan. Hal ini dilakukan dengan mempertimbangkan ketidakpastian yang ada dalam perencanaan pembangkitan, seperti :

- Berapa persen perkiraan pertumbuhan beban per tahun
- Perkiraan ketersediaan dan harga bahan bakar
- Bentuk kurva beban
- Ketersediaan alternatif unit pembangkit baru , dan lain-lain.

Dengan demikian seorang perencana di bidang ini harus mengetahui ruang lingkup yang cukup luas mengenai karakteristik suatu sistem tenaga listrik mencakup ekonomi, biaya dan keandalannya. Hal ini dilakukan dengan memproyeksikan kondisi suatu sistem tenaga listrik beberapa tahun ke depan, selanjutnya mengembangkan beberapa alternatif dan memilih rencana yang optimal diantara alternatif-alternatif tersebut. Dalam proses ini peranan dari perencana sangat dominan, seperti dalam hal penyederhanaan masalah, membuat prioritas rencana dan selanjutnya memutuskan rencana yang paling layak.

Sudah banyak teknik dikembangkan untuk menunjang kegiatan perencanaan pembangkitan ini, seperti penggunaan model matematis untuk memformulasikan permasalahan dalam perencanaan, yang meliputi antara lain : model linier programming, dynamic programming serta model optimisasi. Berdasarkan teknik-teknik tersebut juga sudah banyak program komputer yang ditulis seperti WASP, ZOPPLAN dan lain-lain. Tentunya dengan menggunakan program komputer tersebut, pekerjaan perencanaan pembangkitan menjadi semakin mudah. Akan tetapi semakin rumitnya permasalahan, semakin banyaknya faktor ketidakpastian serta semakin banyaknya kendala yang harus dihadapi, semakin banyak pula pilihan rencana yang harus dipertimbangkan. Sehingga dengan strategi pencarian (untuk memilih rencana yang optimal) saja ternyata tidak cukup untuk menyelesaikan permasalahan perencanaan pembangkitan yang sebenarnya. Secara umum diakui bahwa selain menggunakan program komputer tersebut, faktor pengalaman / pengetahuan para perencana masih memegang peranan yang penting dalam pekerjaan ini. Pengalaman perencana didapat dari kemampuannya untuk menganalisa tingkah laku dan karakteristik suatu sistem tenaga listrik dan dapat membuat keputusan yang lebih tepat.

---

<sup>1</sup> Mahasiswa STT-PLN

<sup>2</sup> Mahasiswa STT-PLN

<sup>3</sup> Dosen Teknik Informatika STT-PLN

Teknik dinamik programming merupakan teknik yang paling banyak digunakan karena memberikan hasil yang mendekati optimal. Akan tetapi teknik ini membutuhkan waktu proses yang cukup lama untuk mendapatkan hasil yang mendekati optimal tersebut. Untuk itu dalam makalah ini akan dibahas teknik untuk menyelesaikan masalah dalam perencanaan pembangkitan dengan menggunakan optimisasi sekuensial yang disederhanakan, sehingga selain waktu prosesnya lebih cepat, akan memberikan hasil yang tidak jauh berbeda dengan teknik dinamik programming.

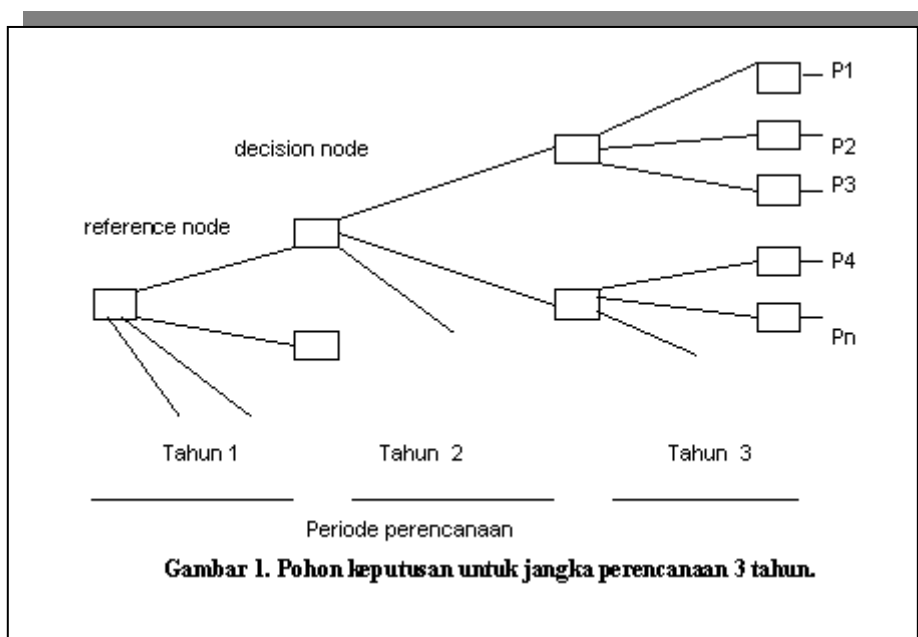
### Teknik Pohon Keputusan dalam Perencanaan Pembangkitan

Lingkup dari kegiatan perencanaan pembangkitan mencakup tiga kelompok variabel, yaitu: variabel keputusan, variabel acak dan variabel atribut. Variabel keputusan adalah kelompok variabel yang dapat dikendalikan oleh perencana, seperti keputusan untuk membangun pusat pembangkit listrik yang baru, menambah kapasitas yang sudah ada atau tidak mengoperasikan lagi suatu unit pembangkit. Variabel acak adalah variabel yang tidak dapat dikendalikan oleh perencana karena adanya faktor ketidakpastian, seperti pertumbuhan beban, harga dan ketersediaan bahan bakar, biaya modal untuk suatu pembangkit baru, tingkat bunga dan lain-lain. Variabel atribut adalah kelompok variabel yang menyatakan tingkat kelayakan dan ketepatan suatu rencana, seperti variabel biaya, variabel dampak lingkungan, variabel pemakaian bahan bakar dan variabel keandalan.

Semua variabel keputusan dalam perencanaan pembangkitan dapat dimodelkan dalam suatu pohon keputusan, yang menggambarkan beberapa keputusan yang tersedia setiap tahun yang selanjutnya untuk dipilih urutan keputusan yang harus dibuat selama jangka waktu perencanaan. Suatu pohon keputusan adalah suatu graph atau tree yang memiliki sifat-sifat sebagai berikut:

- Pohon tersebut memiliki simpul (node) awal yang disebut root
- Memiliki beberapa simpul akhir yang tidak memiliki simpul lanjutan (successor) yang disebut leaf
- Setiap simpul selain root dan leaf, pasti memiliki satu simpul pendahulu (predecessor) dan beberapa simpul lanjutan.

Pohon keputusan ini adalah untuk menyatakan secara terpisah setiap jalur yang mungkin dalam proses perencanaan, yang akan menghasilkan urutan simpul keputusan. Simpul akar menyatakan keadaan sistem pembangkitan sekarang pada tahun ke-0. Simpul lanjutan dari simpul akar menyatakan keputusan pada tahun ke-1 yang paling layak dan diletakkan pada urutan berikutnya sebagai cabang dari simpul akar. Hal ini diteruskan untuk simpul yang lain yang merupakan keputusan pada tahun ke-2, 3 dan seterusnya sampai mencapai simpul daun yang merupakan akhir dari periode perencanaan, sehingga diperoleh urutan keputusan dari simpul akar sampai simpul daun. Gambar 1 memperlihatkan pohon keputusan dalam perencanaan pembangkitan selama jangka waktu 3 tahun.



## Metode Perencanaan Pembangkitan dengan Optimisasi Sekuensial yang Disederhanakan

Dalam metode ini yang dipakai sebagai variabel keputusan adalah jenis-jenis alternatif pilihan pembangkit baru dan maksimum jumlahnya pertahun yang diperbolehkan untuk dibangun. Setiap alternatif dinyatakan sebagai satu simpul. Jumlah simpul setiap tahun adalah sama dengan jumlah alternatif ditambah satu ( yaitu alternatif tidak menambah pembangkit baru ).

Pada setiap tahun, pemilihan alternatif menggunakan Lifecycle Benefit/ Cost ratio sehingga untuk setiap penambahan alternatif pembangkit baru dihitung Lifecycle Cost dan Lifecycle Benefit nya sebagai berikut :

- Capital cost dari alternatif tersebut = C
- Prosentase disbursment biaya konstruksi pada tahun – j = p(j)
- Eskalasi / tahun dari biaya konstruksi = esc
- Discount rate = dsc
- Jangka waktu konstruksi = n

Maka :

$$\text{Lifecycle Cost} = \sum_{j=1}^n \frac{p(j) * C * [1 + esc]^j}{[1 + dsc]^j}$$

- Production cost tanpa adanya alternatif baru pada tahun – j = P0(j)
- Production cost dengan adanya alternatif baru pada tahun – j = P1(j)
- Umur ekonomis dari alternatif baru tersebut = m

Maka :

$$\text{Lifecycle Benefit} = \sum_{j=1}^m \frac{[P0(j) - P1(j)]}{[1 + dsc]^j}$$

dan

$$\text{BC Ratio} = \frac{\text{Lifecycle Benefit}}{\text{Lifecycle Cost}}$$

Alternatif yang dipilih pada tahun tersebut adalah alternatif dengan B/C Ratio paling tinggi. Dimana untuk alternatif tidak menambah pembangkit baru memiliki B/C Ratio = 1.0

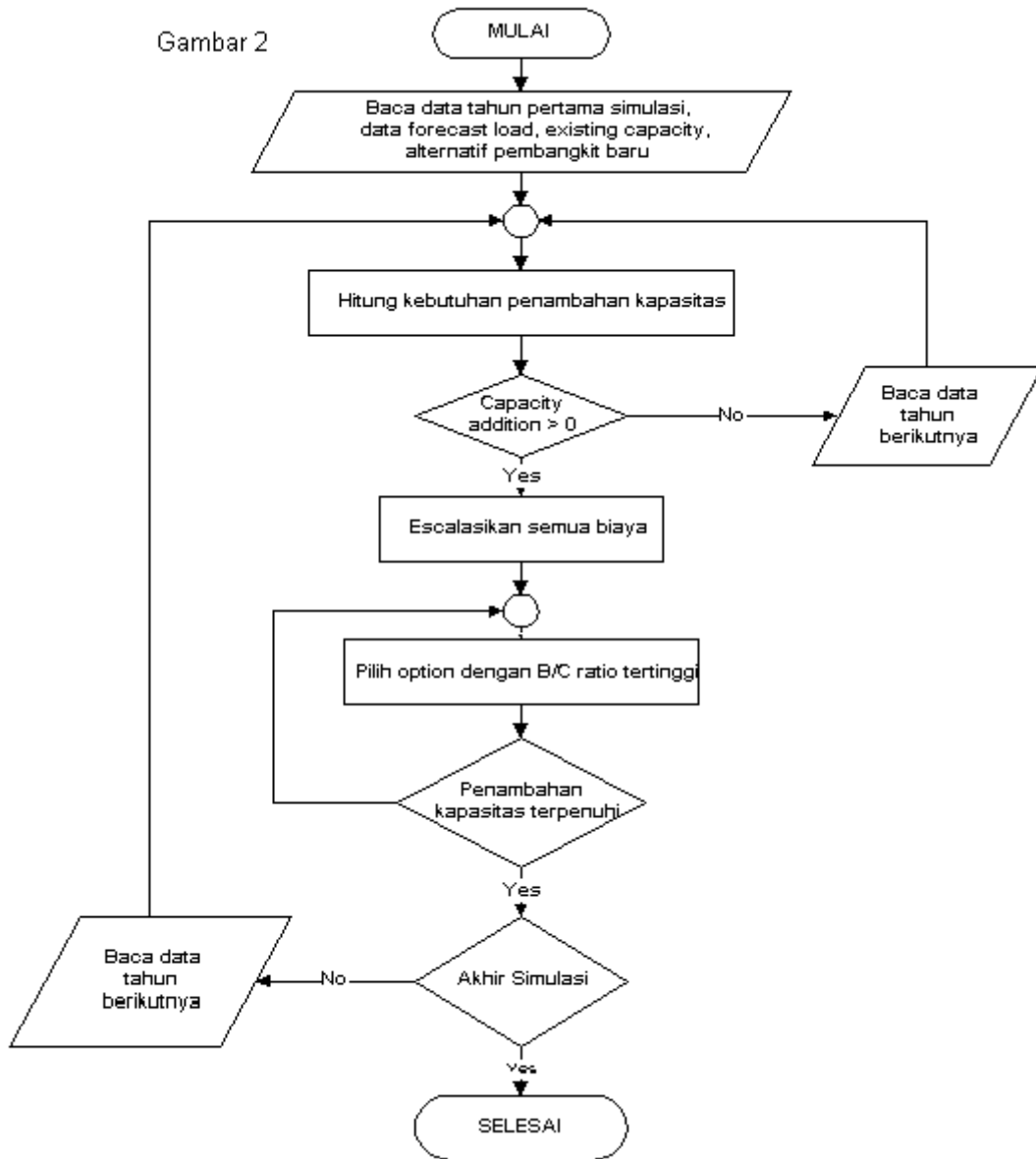
Data pendukung yang dibutuhkan antara lain :

- Beban harian per jam selama perioda simulasi.
- Proyeksi pertumbuhan beban puncak dan energi
- Alternatif-alternatif pembangkit baru, total biaya investasi, lama pembangunan, prosentase disbursment biaya investasi per tahun selama masa pembangunan dan umur ekonomisnya .
- Biaya operasi dan pemeliharaan pertahun dari setiap alternatif tersebut.
- Karakteristik heat rate dari alternatif pembangkit baru tersebut.
- Data pembangkit existing, kapasitas terpasang, karakteristk heat rate dan biaya operasi Tetap dan Variable.

Reserve margin dan komposisi prosentase beban peak, base dan intermediate.

Algoritma perencanaan pembangkitan dengan metoda ini dapat dilihat pada flowchart Gambar 2.

Gambar 2



### Contoh Kasus Perencanaan Pembangkitan

Suatu sistem tenaga listrik terinterkoneksi memiliki daya terpasang pada tahun 1997 sebesar 10.386 MW dengan realisasi beban puncak dan produksi pada tahun tersebut adalah 8.110 MW dan 50.409 GWh. Kurva beban harian dari sistem tersebut dapat dilihat pada Gambar 3, sedangkan komposisi jenis pembangkit pada sistem tersebut seperti terlihat pada Tabel 1 berikut.

| Jenis Pembangkit | Kapasitas Terpasang (MW) | Tipe         |
|------------------|--------------------------|--------------|
| PLTU Batubara    | 2400                     | Base         |
| PLTU Gas         | 800                      | Base         |
| PLTU BBM         | 790                      | Intermediate |
| PLTGU Gas        | 3267                     | Base         |
| PLTGU BBM        | 329                      | Intermediate |
| PLTA Waduk/ROR   | 2026                     | Peak/ Base   |
| PLTP             | 305                      | Base         |
| PLTG             | 377                      | Peak         |
| PLTD             | 92                       | Peak         |

Tabel 1. Komposisi pembangkit existing

Proyeksi pertumbuhan beban dan energi untuk 10 tahun ke depan ( 1998 – 2007 ) untuk tiga skenario Rendah, Menengah dan Tinggi adalah seperti terlihat pada Tabel 2 berikut.

| Tahun | Skenario Pertumbuhan ( % ) |          |        |
|-------|----------------------------|----------|--------|
|       | Rendah                     | Menengah | Tinggi |
| 1998  | 11.0                       | 12.0     | 14.0   |
| 1999  | 10.7                       | 11.8     | 13.5   |
| 2000  | 10.3                       | 11.5     | 13.1   |
| 2001  | 10.0                       | 11.2     | 12.7   |
| 2002  | 9.6                        | 10.7     | 12.4   |
| 2003  | 9.2                        | 10.4     | 11.0   |
| 2004  | 8.7                        | 10.0     | 10.6   |
| 2005  | 8.4                        | 9.7      | 10.2   |
| 2006  | 8.0                        | 9.4      | 9.9    |
| 2007  | 7.5                        | 9.2      | 9.5    |

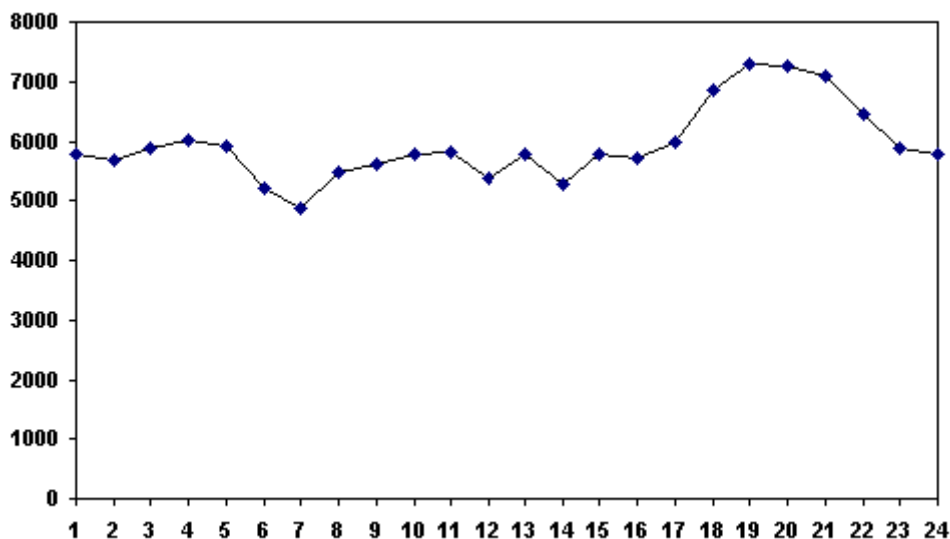
Tabel 2. Skenario Pertumbuhan beban dan energi

Sedangkan alternatif unit pembangkit baru untuk mengisi kebutuhan tambahan kapasitas 10 tahun ke depan adalah seperti terlihat pada Tabel 3 berikut.

| Jenis Pembangkit | Kapasitas (MW) | Kategori Unit | FOR ( % ) | Biaya Modal (\$/ kW) |
|------------------|----------------|---------------|-----------|----------------------|
| PLTU Batubara    | 600            | Base          | 6.0       | 1167                 |
| PLTU Batubara    | 400            | Base          | 6.0       | 1285                 |
| PLTGU Gas        | 500            | Intermediate  | 3.3       | 750                  |
| PLTG HSD         | 130            | Peaking       | 3.3       | 450                  |

Tabel 3. Alternatif pembangkit baru

Dengan batasan reserve margin 25 % - 30 % dan komposisi prosentase beban peak, intermediate dan base adalah 20 %, 30 % dan 50 %, maka dari hasil simulasi, alternatif pembangkit baru yang terpilih untuk memenuhi kebutuhan tambahan kapasitas pada tahun 1998 sampai dengan 2007 adalah seperti pada Tabel 4 berikut.



| Tahun | Skenario |   |   |   |            |          |   |   |   |            |        |   |   |   |            |
|-------|----------|---|---|---|------------|----------|---|---|---|------------|--------|---|---|---|------------|
|       | Rendah   |   |   |   |            | Menengah |   |   |   |            | Tinggi |   |   |   |            |
|       | A        | B | C | D | LOLH (jam) | A        | B | C | D | LOLH (jam) | A      | B | C | D | LOLH (jam) |
| 1998  | 2        | 0 | 0 | 1 | 8.90       | 2        | 0 | 0 | 1 | 11.63      | 2      | 1 | 0 | 0 | 9.76       |
| 1999  | 0        | 3 | 0 | 0 | 15.07      | 1        | 2 | 0 | 1 | 12.30      | 0      | 4 | 0 | 0 | 14.39      |
| 2000  | 0        | 3 | 0 | 1 | 11.58      | 1        | 2 | 0 | 1 | 10.00      | 2      | 0 | 1 | 1 | 10.80      |
| 2001  | 1        | 1 | 1 | 0 | 8.94       | 0        | 2 | 1 | 2 | 8.51       | 2      | 0 | 1 | 2 | 8.65       |
| 2002  | 1        | 2 | 1 | 0 | 7.82       | 2        | 0 | 1 | 1 | 6.64       | 1      | 2 | 1 | 2 | 6.74       |
| 2003  | 0        | 2 | 1 | 0 | 10.86      | 0        | 2 | 2 | 0 | 6.11       | 0      | 3 | 2 | 0 | 4.79       |
| 2004  | 1        | 0 | 1 | 3 | 15.66      | 0        | 2 | 1 | 2 | 13.25      | 1      | 0 | 1 | 4 | 13.67      |
| 2005  | 1        | 0 | 2 | 3 | 11.05      | 1        | 1 | 2 | 2 | 11.65      | 2      | 0 | 2 | 3 | 11.81      |
| 2006  | 1        | 1 | 1 | 1 | 11.28      | 3        | 0 | 1 | 3 | 4.97       | 3      | 0 | 1 | 5 | 5.37       |
| 2007  | 1        | 1 | 1 | 4 | 5.92       | 2        | 0 | 1 | 4 | 4.24       | 2      | 0 | 2 | 0 | 9.05       |

Tabel 4. Tambahan pembangkit baru

Keterangan :

- A = PLTU Batubara 600 MW
- B = PLTU Batubara 400 MW
- C = PLTGU Gas 500 MW
- D = PLTG HSD 130 MW

Dalam model perencanaan pembangkitan yang lain, umumnya untuk menseleksi tambahan pembangkit baru digunakan kriteria LOLP (Loss Of Load Probability). Sebagai contoh ditentukan LOLP sistem sebesar 1 hari /tahun. Sedangkan pada metode optimisasi sekuensial, kriteria tersebut merupakan keluaran dari program, yaitu dalam bentuk LOLH (Loss Of Load Hours). Dari hasil simulasi tersebut diperoleh LOLH setiap tahun < 24 jam dengan kata lain LOLP < 1 hari/ tahun.

## LAMPIRAN PROGRAM DALAM PASCAL

```
Program konversi;
uses crt,dos,printer;
Type Semt                                = array [0..50,0..50] of integer;
    larik                                 =array [0..100] of integer;
    flarik                                = file of larik;
var N,jumlahdata,time,y                   : integer;
    Q,b,P,a                               : array[0..100] of integer;
    jlh,jum,Power                         : array[0..100] of integer;
    i,j,k,t,rek,m,temp,temp1             : integer;
    Prob,Kemunk,sigprob,Terjadi           : array [0..100] of real;
    temp2,temp3,s,LOLP                   : real;
    Memp                                  : Semt;
    pukul,beban                           : larik;
    fnilai                                 : flarik;

Procedure Bacadata(jumlah :integer);
var v,x : integer;
begin
writeln('Masukkan data beban yg diperlukan tiap waktu( dlm mega watt): ');
    writeln (lst,'Masukkan data beban yg diperlukan tiap waktu:');
    v := 0;
    while v <= (jumlah-1) do
begin
write('pukul : ');
write(lst,'pukul : ');readln(pukul[v]);write(lst,pukul[v]);
write(' ', 'beban : ');
write(lst,' ', 'beban : ');readln(beban[v]);
write(lst,beban[v]);
v:=v+1;
writeln(lst);
end;
end;
{ Procedure Tulisdata(jumlah:integer);
var v : integer;
begin
clrscr;
for v := 1 to jumlah do
begin
write(pukul[v],' ');
writeln(beban[v],' ');
end;
end;
Procedure simpandata( var data1:flarik;jumlah :integer);
var v : integer;
begin
Assign ( data1,'Nilai.dat');
Rewrite ( data1);
write(data1,pukul,beban );
end;
Procedure Bacafile(var data2 :flarik;jumlah : integer);
var v : integer;
begin
clrscr;
Reset ( data2);
while NOT EOF (data2) do
begin
read ( data2, pukul,beban);
for v := 1 to jumlah do
begin
write( pukul[v],' ',beban[v]);
```

```

                writeln;
            end;
        end;
    end;
end;
Procedure sorting (jumlah : integer);
var j,k,temp1,temp2 : integer;
begin
    for j := 1 to (jumlah-1) do
        for k := j+1 to jumlah do
            if beban[j] >= beban[k] then
                begin
                    temp1 := pukul[j];
                    pukul[j] := pukul[k];
                    pukul[k] := temp1;
                    temp2 := beban[j];
                    beban[j] := beban[k];
                    beban[k] := beban[j];
                end;
            end;
        end;
    end;
begin
    write ('jumlah data:'); readln( jumlahdata);
    bacadata(jumlahdata);
    Tulisdata(jumlahdata);
    sorting(jumlahdata);
    simPandata(fnilai, jumlahdata);
    bacafile(fnilai, jumlahdata);
end.}
function Waktu( var x: integer):integer;
var v : integer;
begin
    time := 0;
    for v := 0 to (jumlahdata-1) do
        begin
            if jum[x] < beban[v] then
                begin
                    time := time + 1;
                    Waktu := time;
                    y := time;
                end
            else
                begin
                    Waktu := time;
                    y := time;
                end;
            end;
        end;
    end;
end;
Procedure Urut (var Mengu : Semt );
var Simp :Semt;
begin
    for k := (N-1) downto 0 do
        begin
            Simp[i,k ] := Mengu[i,k];
            Mengu[i,k] := Mengu[j,k];
            Mengu[j,k] := Simp[i,k];
        end;
    end;
end;
Procedure Tulis ( Coba : Semt );
begin
    for i := 0 to (rek-1) do
        begin

```

```

        for k := (N-1) downto 0 do
        begin
            write (lst,Coba[i,k]:5 )
            end;
        writeln(lst);
    end;
end;
function satu (t : integer ):real;
var u : integer;
begin
    s := 0.0;
    for u :=t downto 0 do s := s + sigprob[u];
    satu := s;
    LOLP := s * y;
end;
begin
    clrscr;
    write('Masukkan berapa jam data yg diteliti:');
    write(lst,'Masukkan berapa jam data yg diteliti :');read(jumlahdata);
    writeln(lst,' ',jumlahdata);
    Bacadata(jumlahdata);
    write('Masukkan jumlah pembangkit yang digunakan:');
    write(lst,'Masukkan jlh pemb yg digunakan:');readln(N);write(lst,' ',N);
    writeln;
    write ('Masukkan brp mega watt pwr msng2 unit: ');writeln;writeln(lst);
    write(lst,'Masukkan brp mega watt pwr msng2 unit :'); writeln(lst);
    write('Berapa prob Unit pemb.tdk berkerja dgn baik? -->Prob:');
    writeln;
    for m := 0 to (N-1) do
    begin
        write('Unit(',m,')');
        write(lst,'Unit(',m,')');
        read (Power[m]);write(lst,' ',Power[m]);
        write('Prob(',m,')');
        write (lst,' ', 'Prob(',m,')');
        read (Prob[m]);write(lst,' ',Prob[m]);
        readln;
        writeln(lst);
    end;
    writeln (lst,' 1=Power On    0= Power off ');
    writeln (lst,'-----');
    rek := 1;
    for t := 0 to (N-1) do rek := rek * 2;
    for i := 0 to (rek-1) do
    begin
        Q[0] := i div 2;
        b[0] := i - 2*Q[0];
        j := 0;
        while Q[j]<>0 do
        begin
            inc(j);
            Q[j] := ( round(Q[j-1]) div 2 );
            b[j] := Q[j-1] - 2*Q[j];
        end;
        jlh[i] := 0; jum[i] := 0; sigprob[i]:=0;Terjadi[i]:=1.0;
        for k := (N-1) downto 0 do
        begin
            jlh[i]:= jlh[i] + b[k];
            jum[i]:= jum[i] + b[k] * Power[k];
            if b[k] =0 then kemunk[k] := Prob[k]
            else

```

```

        kemunk[k] := (1- Prob[k]);
        Terjadi[i] := Terjadi[i] * kemunk[k];
        Memp[i,k] := b[k];
    end;
    sigprob[i] := sigprob[i] + Terjadi[i];
end;
for i := 0 to (rek-2) do
    for j := (i+1) to ( rek-1) do
        begin
            if jum[i] > jum[j] then
                begin
                    Urut(Memp);
                    temp := jum[i];
                    jum[i] := jum[j];
                    jum[j] := temp;
                    temp1 := jlh[i];
                    jlh[i] := jlh[j];
                    jlh[j] := temp1;
                    temp2 := Terjadi[i];
                    Terjadi[i] := Terjadi[j];
                    Terjadi[j] := temp2;
                    temp3 := sigprob[i];
                    sigprob[i] := sigprob[j];
                    sigprob[j] := temp3;
                end;
            end;
        Tulis(Memp);
        writeln(lst,'-----');
        writeln(lst,' Jumlah          Jumlah          ');
        writeln (lst,' Power          Power          Indivi      Kumul
        writeln (lst,' yang On      Tersedia      Prob.      Prob.      waktu  LOLP ');
        writeln (lst,'-----');
            for i := 0 to (rek-1) do
                begin
                    k := i;
                    write (lst,jlh[i]);
                    write (lst,'          ',jum[i]);
                    write (lst,'          ',Terjadi[i]:8:7);
                    write (lst,'          ',satu(k):8:7);
                    write (lst,'          ',Waktu(k));
                    write (lst,'          ',LOLP:8:7);
                    writeln(lst);
                end;
            writeln(lst);
        end.

```

## LAMPIRAN PROGRAM DALAM VISUAL BASIC

### Form Utama

Option Explicit

Private Type arr

    semt(50, 50) As Integer

    larik(50) As Integer

End Type

Dim n, jumlahData, rek, t, i, j, k As Integer

Dim Q(100), b(100), jlh(100), jum(100), power(100) As Integer

Dim pukul As arr 'larik

Dim beban As arr 'larik

Dim Memp As arr 'semt

Dim temp, temp1, temp2, temp3, time, Y, m As Integer

Dim Prob(2048), sigprob(2048), Terjadi(2048), Kemunk(2048) As Double

Dim P(2048), a(2048) As Integer

Dim s, LOLP As Double

Dim str As String

Private Sub cmdOk\_Click()

    Call proses

    Form2.Show

End Sub

Private Sub proses()

    On Error Resume Next

    With Form2.rtb

        .Text = "

=====  
=====

        & " PROGRAM KONVERSI PEMBANGKIT" & vbCrLf \_

        & " PT PLN (PERSERO)" & vbCrLf \_

        & "

=====  
=====

        & " " & vbCrLf

    .Text = .Text & " JUMLAH JAM YANG DITELITI = " & jumlahData & vbCrLf

    For i = 0 To jumlahData - 1

        .Text = .Text & " Pukul [" & i + 1 & "] : " & pukul.larik(i) & Space(5) & "

Beban [" & i + 1 & "] : " & beban.larik(i) & vbCrLf

    Next i

    .Text = .Text & " " & vbCrLf

    .Text = .Text & " JUMLAH PEMBANGKIT YANG DIGUNAKAN = " & n & vbCrLf

    For i = 0 To n - 1

        .Text = .Text & " Unit [" & i + 1 & "] : " & power(i) & Space(5) & "

Probabilitas [" & i + 1 & "] : " & Format(Prob(i), "#####0.#####0") & vbCrLf

    Next i

    .Text = .Text & " " & vbCrLf

    .Text = .Text & " -----

-----" & vbCrLf

```

Dim hh, xx As Integer
xx = Val(txtPembangkit.Text)

For hh = xx To 1 Step -1
    .Text = .Text & " " & "UNIT" & Space(1)
Next hh

.Text = .Text & " " & vbCrLf

Dim ii, yy As Integer
yy = Val(txtPembangkit.Text)
For ii = yy To 1 Step -1
    .Text = .Text & " " & "[" & ii & "]" & Space(2.5)
Next ii

    .Text = .Text & Space(5) & "X" & Space(5) & "Y" & vbCrLf _
        & " -----" & vbCrLf
-----" & vbCrLf

```

End With

```

rek = 1
For t = 0 To (n - 1)
    rek = rek * 2
Next t

For i = 0 To (rek - 1)
    Q(0) = i \ 2
    b(0) = i - 2 * Q(0)
    j = 0
    While Q(j) > 0
        j = j + 1
        Q(j) = Round(Q(j - 1)) \ 2
        b(j) = Q(j - 1) - 2 * Q(j)
    Wend

    jlh(i) = 0
    jum(i) = 0
    sigprob(i) = 0
    Terjadi(i) = 1#

    For k = (n - 1) To 0 Step -1
        jlh(i) = jlh(i) + b(k)
        jum(i) = jum(i) + b(k) * power(k)
        If b(k) = 0 Then
            Kemunk(k) = Prob(k)
        Else
            Kemunk(k) = (1 - Prob(k))
        End If
        Terjadi(i) = Terjadi(i) * Kemunk(k)
        Memp.semt(i, k) = b(k)
    Next k
    sigprob(i) = sigprob(i) + Terjadi(i)
Next i

For i = 0 To (rek - 2)
    For j = (i + 1) To (rek - 1)
        If jum(i) > jum(j) Then
            Call Urut(Memp)
            temp = jum(i)
            jum(i) = jum(j)

```

```

        jum(j) = temp
        temp1 = jlh(i)
        jlh(i) = jlh(j)
        jlh(j) = temp1
        temp2 = Terjadi(i)
        Terjadi(i) = Terjadi(j)
        Terjadi(j) = temp2
        temp3 = sigprob(i)
        sigprob(i) = sigprob(j)
        sigprob(j) = temp3
    End If
Next j
Next i

Call Tulis(Memp)

Form2.rtb.Text = Form2.rtb.Text & " -----"
-----" & vbCrLf & _
        " KETERANGAN INDIKATOR : " & vbCrLf & _
        " 1 = POWER ON          ; 0 = POWER OFF" & vbCrLf & _
        " X = JUMLAH POWER ON   ; Y = JUMLAH POWER TERSEDIA [MW]" &
vbCrLf & _
        " " & vbCrLf & _
        " " & vbCrLf

With Form2.rtb
    .Text = .Text & " -----"
-----" & vbCrLf & _
        " JUMLAH POWER    JUMLAH POWER    INDIVI            KUMUL        WAKTU
" & vbCrLf & _
        " YANG ON          TERSEDIA [MW]    PROB.            PROB.        [JAM]
LOLP " & vbCrLf & _
        " -----"
-----" & vbCrLf

For i = 0 To (rek - 1)
    k = i
    .Text = .Text & " " & Format(jlh(i), "##0") & Space(15) & _
        " " & Format(jum(i), "##0") & Space(13) & _
        " " & Format(Terjadi(i), "#####0.#####0") & Space(5) & _
        " " & Format(Satu(k), "#####0.#####0") & Space(4) & _
        " " & Waktu(k) & Space(7) & _
        " " & Format(LOLP, "#####0.#####0") & vbCrLf
Next i
End With

Form2.rtb.Text = Form2.rtb.Text & " -----"
-----" & vbCrLf & _
        " " & vbCrLf

On Error GoTo 0
End Sub

Private Function Waktu(ByRef X As Integer) As Integer
    Dim v As Integer

    time = 0
    For v = 0 To (jumlahData - 1)
        If jum(X) < beban.larik(v) Then
            time = time + 1
        End If
    Next v
End Function

```

```

        Waktu = time
        Y = time
    Else
        Waktu = time
        Y = time
    End If
Next v
End Function

Private Sub Urut(ByRef Mengu As arr) 'semt
    Dim Simp As arr 'semt

    For k = (n - 1) To 0 Step -1
        Simp.semt(i, k) = Mengu.semt(i, k)
        Mengu.semt(i, k) = Mengu.semt(j, k)
        Mengu.semt(j, k) = Simp.semt(i, k)
    Next k
End Sub

Private Sub Tulis(coba As arr) 'semt
    Dim tem, jlh As Integer

    For i = 0 To (rek - 1)
        For k = (n - 1) To 0 Step -1
            Form2.rtb.Text = Form2.rtb.Text & " " & coba.semt(i, k) & Space(4)

            If coba.semt(i, k) = 1 Then jlh = jlh + 1

        Next k
        Form2.rtb.Text = Form2.rtb.Text & Space(5) & jlh & Space(5) & jum(i) & vbCrLf
        jlh = 0
    Next i
End Sub

Private Function Satu(t As Integer) As Double
    Dim u As Integer

    s = 0#
    For u = t To 0 Step -1
        s = s + sigprob(u)
    Next u
    Satu = s
    LOLP = s * Y
End Function

Private Sub Form_Load()
    m = 1
    i = 0
End Sub

Private Sub txtJam_KeyPress(KeyAscii As Integer)
    On Error Resume Next
    jumlahData = Val(txtJam.Text)
    If (KeyAscii = 13) And (m < jumlahData) Then
        lblPkl.Caption = "Pukul " & "[" & m & "]"
        lblBeban.Caption = "Beban [MW] " & "[" & m & "]"
        txtPkl.SetFocus
    End If
    On Error GoTo 0
End Sub

```

```

Private Sub txtPkl_KeyPress(KeyAscii As Integer)
    On Error Resume Next
    If (KeyAscii = 13) And (m <= jumlahData) Then
        pukul.larik(i) = Val(txtPkl.Text)
        txtBeban.SetFocus
    End If
    On Error GoTo 0
End Sub

Private Sub txtBeban_KeyPress(KeyAscii As Integer)
    On Error Resume Next
    If KeyAscii = 13 And (m <= jumlahData) Then
        beban.larik(i) = Val(txtBeban.Text)
        m = m + 1
        i = i + 1
        If Not (m = jumlahData + 1) Then
            txtPkl.Text = ""
            txtBeban.Text = ""
            lblPkl.Caption = "Pukul " & "[" & m & "]"
            lblBeban.Caption = "Beban [MW] " & "[" & m & "]"
        End If
        txtPkl.SetFocus
    End If
    If KeyAscii = 13 And (m = jumlahData + 1) Then
        txtPembangkit.SetFocus
    End If
    On Error GoTo 0
End Sub

Private Sub txtPembangkit_KeyPress(KeyAscii As Integer)
    n = Val(txtPembangkit.Text)
    m = 1
    i = 0
    If (KeyAscii = 13) And (m < n) Then
        lblJml.Caption = "Jumlah [MW] " & "[" & m & "]"
        lblProb.Caption = "Probabilitas Forced Outage Rate " & "[" & m & "]"
        txtJml.SetFocus
    End If
    On Error GoTo 0
End Sub

Private Sub txtJml_KeyPress(KeyAscii As Integer)
    On Error Resume Next
    If (KeyAscii = 13) And (m <= n) Then
        power(i) = Val(txtJml.Text)
        txtProb.SetFocus
    End If
    On Error GoTo 0
End Sub

Private Sub txtProb_KeyPress(KeyAscii As Integer)
    On Error Resume Next
    If KeyAscii = 13 And (m <= n) Then
        Prob(i) = Val(txtProb.Text)
        m = m + 1
        i = i + 1
        If Not (m = n + 1) Then
            txtJml.Text = ""
            txtProb.Text = ""
            lblJml.Caption = "Jumlah [MW] " & "[" & m & "]"
            lblProb.Caption = "Probabilitas Forced Outage Rate " & "[" & m & "]"
        End If
    End If
    On Error GoTo 0
End Sub

```

```

        End If
        txtJml.SetFocus
    End If
    If KeyAscii = 13 And (m = n + 1) Then
        cmdOk.SetFocus
    End If
    On Error GoTo 0
End Sub

Private Sub cmdOk_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
    MenuUtama.Status.Panels(1).Text = " Klik tombol untuk melakukan proses perhitungan..."
End Sub

Private Sub cmdBatal_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
    MenuUtama.Status.Panels(1).Text = " Klik tombol untuk membatalkan proses perhitungan..."
End Sub

Private Sub cmdKeluar_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
    MenuUtama.Status.Panels(1).Text = " Klik tombol untuk keluar dari form..."
End Sub

Private Sub cmdBatal_Click()
    On Error Resume Next
    txtJam.Text = ""
    txtPkl.Text = ""
    txtBeban.Text = ""
    txtPembangkit.Text = ""
    txtJml.Text = ""
    txtProb.Text = ""
    lblPkl.Caption = "Pukul "
    lblBeban.Caption = "Beban [MW] "
    lblJml.Caption = "Jumlah [MW] "
    lblProb.Caption = "Probabilitas Forced Outage Rate "
    i = 0
    m = 1
    txtJam.SetFocus
    On Error GoTo 0
End Sub

Private Sub cmdKeluar_Click()
    Unload Me
End Sub

```

## SCREEN SHOT

Program LOLP

PT PLN

Jumlah Jam Yang Diteliti :

Masukkan data beban yg diperlukan tiap waktu [MW]

Pukul   
Beban [MW]

---

Masukkan jumlah pembangkit yang digunakan

Jumlah [MW]   
Probabilitas Forced Outage Rate

**Proses** **Batal** **Keluar**

Form Utama

```

PROGRAM LOLP
PT PLN (PERSERO)
=====
JUMLAH JAM YANG DITELITI = 2
Pukul [1] : 2      Beban [1] : 5
Pukul [2] : 3      Beban [2] : 9

JUMLAH PEMBANGKIT YANG DIGUNAKAN = 2
Unit [1] : 4      Probabilitas [1] : 0.6000000
Unit [2] : 8      Probabilitas [2] : 0.9000000
=====
UNIT  UNIT
[2]  [1]      X      Y
-----
0    0         0      0
0    1         1      4
1    0         1      8
1    1         2     12
=====
KETERANGAN INDIKATOR :
1 = POWER ON      ; 0 = POWER OFF
X = JUMLAH POWER ON ; Y = JUMLAH POWER TERSEDIA [MW]
=====
JUMLAH POWER  JUMLAH POWER  INDIVI    KUMUL    WAKTU
YANG ON       TERSEDIA [MW]  PROB.     PROB.    [JAM]    LOLP
-----
0              0              0.5400000  0.5400000  2        0.0000000
    
```

**Cetak Ke Printer**

Form Laporan

**KESIMPULAN**

Program VB ini hanya dapat menampung jumlah pembangkit < 10 pembangkit. Masih bermasalah dalam alokasi memori pada Array. Mohon pencerahannya bagi pembaca sekalian.

**KIRIMKAN KE :**

yogapln@yahoo.com

<http://prihastomo.wordpress.com>

**DAFTAR PUSTAKA**

<http://www.elektroindonesia.com/elektro/ener24.html>