

**LAMPIRAN**

Seed yang dipanggil oleh prosedur randdf pada pembangkitan bilangan random uniform sebanyak 100 data.

1973272912; 281629770; 20006270;  
1280689831; 2096730329; 1933576050;  
913566091; 246780520; 1363774876;  
604901985; 1511192140; 1259851944;  
824064364; 150493284; 242708531;  
75253171; 1964472944; 1202299975;  
233217322; 1911216000; 726370533;  
403498145; 993232223; 1103205531;  
762430696; 1922803170; 1385516923;  
76271633; 413682397; 726466604;  
336157058; 1432650381; 1120463904;  
595778810; 877722890; 1046574445;  
68911991; 2088367019; 748545416;  
622401386; 2122378830; 640690903;  
1774806513; 2132545692; 2079249579;

78130110; 852776735; 1187867272;  
1351423507; 1645973084; 1997049139;  
922510944; 2045512870; 898585771;  
243649545; 1004818771; 773686062;  
403188473; 372279877; 1901633463;  
498067494; 2087759558; 493157915;  
597104727; 1530940798; 1814496276;  
536444882; 1663153658; 855503735;  
67784357; 1432404475; 619691088;  
119025595; 880802310; 176192644;  
1116780070; 277854671; 1366580350;  
1142483975; 2026948561; 1053920743;  
786262391; 1792203830; 1494667770;  
1923011392; 1433700034; 1244184613;  
1147297105; 539712780; 1545929719;  
190641742; 1645390429; 264907697;  
620389253; 1502074852; 927711160;  
364849192; 2049576050; 638580085;  
547070247;

**Lampiran Implementasi Algoritma Monte Dalam Perancangan Filter Pasif :**

```
Program SN_Montec;
```

```
uses crt,dos,normdist,randunfm;
```

```
const
```

```
log_e=0.4342944819;
```

```
var
```

```
sum,X,Y,SNWC,Wc,D,Rs,Gsen,Rg,Vs,R2,C,Count,Sm,V,makr3,maksn,R3 : real;
```

```
makrs,makgsen,makrg,makvs,makc,makr2 : real;
```

```
n,i,j : integer;
```

```
detal1001,dawal100,jawal,dawal,menital,jam,menit,detik,detik100 : word;
```

```
s1,s2,s3,s4,jam1,menit1,detik1,detik1001 : word;
```

```
procedure hilang_kursor;
```

```
var
```

```
reg : registers;
```

```
begin
```

```
reg.ah := 01;
```

```
reg.ch := 01;
```

```
reg.cl := 00;
```

```
intr($10,reg);
```

```
end;
```

Procedure HitungRS;

begin

sum:=0;

for k:=1 to 12 do

begin

RS:=rand(1);

sum:=sum+Rs;

end;

Rs:=sum-6;

Rs:=120+0.3596708\*Rs;

end;

Procedure HitungRG;

begin

sum:=0;

for k:=1 to 12 do

begin

Rg:=rand(1);

sum:=sum+Rg;

end;

Rg:=sum-6;

Rg:=98+0.29398775824\*Rg;

end;

Procedure HitungGSEN;

begin

sum:=0;

for k:=1 to 12 do

begin

Gsen:=rand(1);

sum:=sum+Gsen;

end;

Gsen:=sum-6;

Gsen:=657.58+1.9720009\*Gsen;

end;

Procedure HitungVS;

begin

sum:=0;

for k:=1 to 12 do

begin

VS:=rand(1);

sum:=sum+VS;

end;

Vs:=sum-6;

Vs:=0.015+0.00045033\*Vs;

end;

Procedure inialisasi;

begin

D:=3.00;

Wc:=6.84;

end;

Procedure model;

begin

$R2 := (Vs * Rg * Rs - D * Gsen * R3 * Rs) / (D * Gsen * (R3 + Rs));$

$C := (Vs * Rg * (R3 + Rs)) / (2 * pi * Wc * R3 * (abs(Vs) * Rg * Rs - D * Gsen * R3 * Rs));$

$Wc := (((R2 + Rg) * (Rs + R3) + R3 * Rs) / (2 * pi * (R2 + Rg) * R3 * Rs * C));$

end;

Procedure jumlah\_kumulatif;

begin

Y:=Y+Wc;

X:=X+Wc\*Wc;

end;

Procedure hitung\_sn;

begin

$S_m := (Y * Y) / 100;$

$V := (X - S_m) / (99);$

textbackground(blue);

textcolor(yellow);

```
if ((Sm-V)/(100*V) < 0 then
SNWC:=10*log_e*ln(abs(((Sm-V)/(100*V))))
else
SNWC:=10*log_e*ln((((Sm-V)/(100*V)))));
if (snwc > maksn) then
begin
maksn:=snwc;
makr3:=R3;
end;

gotoxy(30,12);
writeln('R3 = ',R3:8:2,' ';SN = ',SNWC:4:2);
end;
```

```
{Program Utama }
```

```
begin
clrscr;
textbackground(blue);
gettime(jam,menit,detik,detik100);
jawal :=jam;
menital:=menit;
dawal:=detik;
deta100:=detik100;
textcolor(yellow+ blink);
gotoxy(33,5);
```

```
writeln('Waktu Awal Proses : ');  
textcolor(white);  
gotoxy(35,6);  
writeln(jam,' : ',menit,' : ',detik,'.',detik100);  
hilang_kursor;  
R3:=1;  
randdf;  
makrs:=0;  
makrg:=0;  
makgsen:=0;  
makvs:=0;  
makr2:=0;  
makr3:=0;  
maksn:=0;  
repeat  
begin  
X:=0;Y:=0;  
for i:=1 to 100 do  
begin  
HitungRS;  
HitungRG;  
HitungGSEN;  
HitungVS;  
inisialisasi;  
model;
```



```
jumlah_kumulatif;
end;
    hitung_sn;
    R3:=R3+10;
end;
until R3 > 625;
    gettime(jam1,menit1,detik1,detik1001);
    gotoxy(33,18);
    writeln("Waktu Akhir Proses : ");
    textcolor(white);
    gotoxy(35,19);
    writeln(jam1,' : ',menit1,' : ',detik1,',',detik1001);

    if (detik1001 < detail1001) then
        begin
            detik1001:=detik1001+100;
            detik1:=detik1-1;
        end;

    if (detik1 < dawal) then
        begin
            detik1:=detik1+60;
            menit1:=menit1-1;
        end;
end;
```

```
if (menit1 < menital) then
  begin
    menit1:=menit1+60;
    jam1:=jam1-1;
  end;

s1:=abs(jam1-jawal);
s2:=abs(menit1-menital);
s3:=abs(detik1-dawal);
s4:=(abs(detik1001-detak1001));
textcolor(yellow);
gotoxy(29,20);
writeln('Lama Proses Berlangsung : ');
textcolor(white);
gotoxy(35,21);
write(s1,' : ',s2,' : ',s3,' : ',s4);

gotoxy(30,14);
writeln('SN Maksimal : ',maksn:10:4);
gotoxy(30,15);
writeln('Nilai R3   : ',makr3:10:4);

readln
end.
```

UNIT RANDUNFM;

{ Pembangkit bilangan random dengan metode Linear Congruential generator

$Z[I] = ( 630360016 * Z[I-1] ) ( \text{MOD } 214748647 )$  menurut :

MARSE DAN ROBERT

Catatan :

Terdapat 4 Aturan penggunaan program pembangkit bilangan random yaitu :

1. Sebelum menggunakan pembangkit, gunakan prosedur "randdf" sebagai awalan pembangkitan.
  2. Untuk mendapatkan bilangan random  $U(0,1)$  berikutnya, lakukan  
 $U := \text{rand}(\text{stream})$ .
  3. Untuk memasang benih stream-stream terhadap nilai Zset yang diinginkan, lakukan  
 $\text{Randst}(Zset, \text{stream})$
  4. Untuk mendapatkan bilangan bulat sekarang dalam urutan yang dibangkitkan untuk stream-stream ke dalam variabel integer Zget, lakukan  
 $Zget = \text{Randgt}(\text{stream})$
- }

## INTERFACE

uses crt;

var

zrng : array [1..100] of Longint;

Procedure Randdf;

Function Rand(Stream : Integer):real;

Procedure Randst(Zset : Integer,Stream :Integer);

Function Randgt(Stream : Integer) : Integer;

## IMPLEMENTATION

Procedure Randdf;

{ \$I \tp7\bin\random.lib }

Function Rand; { Pembangkit bilangan random berikutnya }

const

B2E15 = 32768;

B2E16 = 65536;

Modlus = 2147483647;

Mult1 = 24112;

Mult2 = 26143;

var

Hi15,Hi31,Low15,Lowprd,Ovflow,Zi : Longint;

```

begin
  { Pembangkit bilangan random berikutnya }
  Zi :=Zrng[Stream];
  Hi15 :=Zi DIV B2E16;
  Lowprd :=(Zi - Hi15 * B2E16) * Mult1;
  Low15 :=Lowprd DIV B2E16;
  Hi31 :=Hi15 * Mult1 + Low15;
  Ovflo :=Hi31 DIV B2E15;
  Zi :=(((Lowprd - Low15 * B2E16) - Modlus)+
        (Hi31 - Ovflo * B2E15)*B2E16) + Ovflo;
  if Zi < 0 then Zi := Zi + Modlus;
  Hi15 :=Zi DIV B2E16;
  Lowprd :=(Zi - Hi15 * B2E16) * Mult2;
  Low15 :=Lowprd DIV B2E16;
  Hi31 :=Hi15 * Mult2 + Low15;
  Ovflo :=Hi31 DIV B2E15;
  Zi :=(((Lowprd - Low15 * B2E16) - Modlus) +
        (Hi31 - Ovflo * B2E15) * B2E16) + Ovflo;
  if Zi < 0 then Zi := Zi + Modlus;
  Zrng[stream]:=Zi;
  Rand :=(2 * (Zi DIV 256) + 1)/ 16777216.0;
end;

```

```
Procedure Randst;
```

```
begin
```

```
    Zrng[stream]:=Zset;
```

```
end;
```

```
Function Randgt;
```

```
begin
```

```
    Randgt:=Zrng[stream];
```

```
end;
```

```
UNIT NORMDIST;
```

```
{ Pembangkit random Variete berdistribusi Normal ( N(0,1 ) dengan  
menggunakan random generator Uniform ( U(0,1) ) }
```

```
INTERFACE
```

```
uses crt,RANDUNFM;
```

```
Function Climit(n:integer):real;
```