

Rismira Andriyani, S.Kom

Algoritma dan Pemograman PERULANGAN



**Untuk Kelas XI SMA
Semester 2**

**SMA Negeri 3 Magelang
Tahun Pelajaran 2016/2017**

LEMBAR PENGESAHAN

JUDUL: ALGORITMA DAN PEMOGRAMAN (PENGULANGAN)

OLEH :

Nama : Rismira Andriyani, S.Kom
NIP : 19760824 200903 2 003
Pangkat / Golongan : Penata Muda Tingkat I / III. B
Unit Kerja : SMA Negeri 3 Magelang

Magelang, Januari 2015

Mengesahkan,

Kepala SMA Negeri 3 Magelang

Kepala Perpustakaan

Joko Tri Haryanto, S.Pd

NIP. 19641020 198803 1 009

Dra. Dyah Kusumastuti W.

NIP. 19630222 198803 2 007

KATA PENGANTAR

Segala puji bagi Allah S.W.T karena atas limpahan nikmat serta karuniaNya sehingga penulis dapat menyelesaikan Modul “Mengelola Objek dengan Corel Draw X5” tanpa halangan suatu apapun.

Mudah-mudahan modul ini dapat dijadikan bahan pembelajaran untuk siswa kelas XII pada semester 1 tahun pelajaran 2015/2016. Dengan tersusunnya modul ini dapat membantu memudahkan siswa dalam melakukan praktek pelajaran TIK di laboratorium komputer. Selanjutnya dalam kesempatan ini penulis ucapkan terima kasih yang sebanyak-banyaknya kepada:

1. Joko Tri Haryanto, S.Pd selaku Kepala sekolah yang sudah memberikan bantuan, arahan serta masukannya sehingga modul ini dapat tersusun seperti ini.
2. Semua pihak yang telah membantu yang tidak dapat kami sebutkan satu per satu.

Semoga modul ini bermanfaat bagi siswa SMA Negeri 3 Magelang, serta guru-guru lain yang akan mempelajari database atau basis data serta implementasinya.

Magelang, Januari 2015

Penulis

DAFTAR ISI

LEMBAR PENGESAHAN	i
KATA PENGANTAR	iii
DAFTAR ISI	iv
BAB 1. PENDAHULUAN	1
1.1. Deskripsi	1
1.2. Prasyarat	1
1.3. Petunjuk penggunaan Modul	1
1.4. Standar Kompetensi	1
1.5. Kompetensi Dasar	1
1.6. Tujuan Pembelajaran	1
1.7. Cek Kemampuan	1
BAB 2. KEGIATAN BELAJAR	2
2.1. Tujuan	2
2.2. Uraian Materi	2
2.2.1. Struktur Perulangan For	2
2.2.2. Struktur Perulangan While-Do	6
2.2.3. Struktur perulangan Repeat..Until	9
BAB 3. UJI KOMPETENSI	12
3.1. Penilaian kompetensi	12
DAFTAR PUSTAKA	16

BAB 1. PENDAHULUAN

1.1. Deskripsi

Perulangan (*loop*) merupakan bentuk yang sering di temui didalam suatu program aplikasi. Di dalam pascal, dikenal tiga macam perulangan yaitu dengan menggunakan statemen *For*, *While-Do* dan *Repeat .. Until*.

1.2. Prasyarat

Mempelajari materi mengenal pascal ini merupakan syarat untuk dapat melanjutkan ke materi berikutnya yaitu pemograman tingkat lanjut.

1.3. Petunjuk penggunaan Modul

1. Bacalah materi modul dengan baik dan pahami konsep yang disampaikan dalam modul ini
2. Perhatikan dengan baik contoh soal yang telah disampaikan dalam modul
3. Jika ada kesulitan dalam mempelajari modul ini bertanyalah kepada guru mata pelajaran yang menguasai materi ini saat tatap muka atau membaca buku lain untuk membandingkan
4. Kerjakan soal-soal latihan pada modul ini dengan baik.

1.4. Standar Kompetensi

Mengenal Bahasa Pemograman

1.5. Kompetensi Dasar

Pengulangan

1.6. Tujuan Pembelajaran

Setelah mempelajari modul ini, peserta didik diharapkan untuk dapat :

1. Menjelaskan fungsi perulangan
2. Membuat struktur pengulangan dengan benar

1.7. Cek Kemampuan

1. Sebutkan struktur pengulangan yang anda ketahui ?
2. Apa yang dimaksud dengan *nestes loop* itu ?
3. Variable pengulang selalu menggunakan tipe ?
4. Struktur *repeat .. until* digunakan untuk statemen ?
5. Apa perbedaan antara struktur *repeat until* dengan *while do* ?

BAB 2. KEGIATAN BELAJAR

2.1. Tujuan

1. Dapat membuat program perulangan terstruktur dengan benar
2. Dapat membuat perulangan dengan repeat .. until
3. Dapat membuat perulangan dengan for to do
4. Dapat membuat perulangan dengan while .. do

2.2. Uraian Materi

Perulangan (*loop*) merupakan bentuk yang sering di temui didalam suatu program aplikasi. Di dalam pascal, dikenal tiga macam perulangan yaitu dengan menggunakan statemen *For*, *While-Do* dan *Repeat .. Until*.

2.2.1. Struktur Perulangan For

Perulangan dengan statement *For* digunakan untuk mengulang atau satu blok statement berulang kali sejumlah yang ditentukan. Perulangan dengan statement *For* dapat berbentuk perulangan positif, perulangan negative dan perulangan tersarang.

1. Perulangan positif

Perulangan positif adalah perulangan dengan penghitung (*counter*) dari kecil ke besar atau dengan kata lain pertambahannya positif.

Perulangan positif dapat dibentuk dengan menggunakan statemen *FOR – TO – DO*, dengan bentuk umum :

For variable-kontrol := nilai-awal To nilai-akhir Do Statemen

variable-kontrol, nilai-awal, nilai-akhir harus mempunyai tipe sama, yaitu bertipe *integer*.

Contoh :

```
program ulang1;  
  
uses crt;  
  
var i : integer;  
  
begin  
  clrscr;  
  for i := 1 to 5 do  
    writeln('I Love SMANAGA');  
  readln;  
end.
```

Program 2.1 Penggunaan perulangan positif sederhana

Bila program ini di jalankan, akan didapatkan hasil :

I Love SMANAGA
I Love SMANAGA
I Love SMANAGA
I Love SMANAGA
I Love SMANAGA

Penjelasan :

Berarti statemen WriteLn('I Love SMANAGA') akan diulang sebanyak 5 kali yaitu dengan penghitunga i dari nilai-awal1 sampai nilai-akhir 5.

Contoh 2 :

```
program ulang2;  
  
uses crt;  
  
var i : integer;  
  
begin  
  clrscr;  
  for i := 1 to 5 do  
    begin  
      writeln(i, ' I Love SMANAGA');  
    end;  
  readln;  
end.
```

Program 2.2 Penggunaan perulangan positif sederhana dalam blok statemen

Bila program ini di jalankan, akan didapatkan hasil :

1. I Love SMANAGA
2. I Love SMANAGA
3. I Love SMANAGA
4. I Love SMANAGA
5. I Love SMANAGA

Contoh 3:

```
program ulang3;  
  
uses crt;  
  
var x,x2,x3 : integer;  
    seperx : real;  
  
begin  
  clrscr;
```

```

writeln('-----');
writeln(' X   X*X   X*X*X   1/X');
writeln('-----');
for x := 1 to 5 do
begin
  X2 := X*x;
  X3 := x*x*x;
  SeperX := 1/x;
  writeln(X:4,X2:10,X3:10,Seperx:12:3);
end;
writeln('-----');
readln;
end.

```

Program 2.3 Penggunaan perulangan positif sederhana dalam bentuk tabel

Bila program ini di jalankan, akan didapatkan hasil :

X	X*X	X*X*X	1/X
1	1	1	1.000
2	4	8	0.500
3	9	27	0.333
4	16	64	0.250
5	25	125	0.200

2. Perulangan negative

Perulangan negative adalah perulangan dengan penghitung (*counter*) dari besar ke kecil atau dengan kata lain pertambahannya negative. Perulangan negative dapat dibentuk dengan menggunakan statemen *For –Down To – Do*.

For variable-kontrol := nilai-awal Down To nilai-akhir Do Statemen

```

program ulang4;

uses crt;

var i : integer;

begin
  clrscr;
  for i := 5 downto 1 do
  begin
    writeln(i,' I Love SMANAGA');
  end;
end;

```


<pre> end; readln; end.</pre>
Program 2.4 Penggunaan perulangan negative sederhana

Bila program ini di jalankan, akan didapatkan hasil :

5. *I Love SMANAGA*
4. *I Love SMANAGA*
3. *I Love SMANAGA*
2. *I Love SMANAGA*
1. *I Love SMANAGA*

3. Perulangan tersarang

Perulangan tersarang (*nested loop*) adalah perulangan yang berada di dalam perulangan yang lainnya. Perulangan yang lebih dalam akan diproses terlebih dahulu sampai habis, kemudian perulangan yang lebih luar baru akan bertambah, mengerjakan perulangan yang lebih dalam lagi mulai dari nilai awalnya dan seterusnya.

Contoh :

<pre> program ulang5; uses crt; var i,j : integer; begin clrscr; for i := 1 to 5 do begin for j := 1 to 3 do write(i:8, j:3); writeln; end; readln; end. end.</pre>
Program 2.5 Penggunaan perulangan tersarang

Bila program ini di jalankan, akan didapatkan hasil :

```

1 1      1 2      1 3
2 1      2 2      2 3
3 1      3 2      3 3
4 1      4 2      4 3
5 1      5 2      5 3
```

2.2.2. Struktur Perulangan While-Do

1. Perulangan While-Do sederhana

Perulangan ini menggunakan statemen *While-Do* yang mempunyai struktur dalam bentuk diagram sebagai berikut

While → *ungkapan-logika* → *Do* → *Statemen*

Statemen *While-Do* digunakan untuk melakukan proses perulangan suatu statemen atau blok statemen terus menerus selama kondisi ungkapan-logika pada *While* masih bernilai logika benar.

```
program ulang6;
uses crt;
var i : integer;
begin
  clrscr;
  i := 0;
  while I < 5 do
  begin
    writeln(i);
    i := i + 1;
  end;
  readln;
end.
```

Program 2.6 Penggunaan perulangan While-Do sederhana

Bila program ini di jalankan, akan didapatkan hasil :

```
0
1
2
3
4
```

Contoh 7 :

```
program ulang7;
uses crt;

var
  nilai, total, ratarata : real;
  N,I                    : integer;

begin
```

```

clrscr;
{*** Memasukkan jumlah dari data ***}

write('Jumlah data ? ');
readln(N);
writeln;

{* Memasukkan nilai-nilai data dan langsung dihitung nilai totalnya *}

I := 0;
Total := 0;
While I < N Do
begin
  I := I + 1;
  write('Nilai data ke ',I,'? ');
  readln(Nilai);
  Total := Total + Nilai;
end;

{*Menghitung nilai rata-ratanya *}
Ratarata := Total/N;

{* Menampilkan hasil *}
writeln;
writeln('Total Nilai    = ',Total:8:2);
writeln('Jumlah data    = ',N:4);
writeln('Rata rata nilai = ',RataRata:8:2);
readln;
end.

```

Program 2.7 Perulangan While-Do untuk menghitung rata-rata nilai

Bila program ini di jalankan, akan didapatkan hasil :

Jumlah data ? 4

Nilai data ke 1 ? 12.35

Nilai data ke 2 ? 34.50

Nilai data ke 3 ? 30.67

Nilai data ke 4 ? 58.90

Total Nilai = 136.42

Jumlah data = 4

Rata rata = 34.11

2. Perulangan While-Do Tersarang

Perulangan *While-Do* tersarang (*Nested While-Do*) merupakan perulangan *While-Do* satu didalam perulangan *While-Do* yang lain.

Contoh 8

```
program ulang8;

uses crt;

var
  Nilai, Total, ratarata : real;
  N,I                    : integer;
  lagi                  : char;

begin
  lagi := 'Y';
  while lagi = 'Y' Do
  begin
    Clrscr;
    {**** Masukkan jumlah dari data ****}

    write('Jumlah data ? ');
    readln(n);
    writeln;

    { * Memasukkan nilai - nilai data dan langsung dihitung nilai totalnya 8 }

    I := 0;
    Total := 0;
    While I < N Do
    Begin
      I := I + 1;
      write('Nilai data ke ',I,' ');
      readln(nilai);
      Total := Total + Nilai;
    end;

    { * Menghitung nilai rata-ratanya * }

    Ratarata := Total/N;

    { * Menampilkan hasil * }

    writeln;
    writeln('Total nilai      = ',Total:8:2);
    writeln('Jumlah data          = ',N:3);
    writeln('Rata-rata nilai      = ',ratarata:7:2);
    writeln;
```

```

write('Akan menghitung lagi (Y/T) ? ');
readln(lagi);
end;
readln;
end.

```

Program 2.8 penggunaan perulangan While-Do tersarang

Bila program dijalankan :

Jumlah data ? 3

Nilai data ke 1 ? 15.2

Nilai data ke 2 ? 13.57

Nilai data ke 3 ? 18.60

Total nilai = 47.37

Jumlah data = 3

Rata-rata = 15.79

Akan menghitung lagi (Y/T) ? T

2.2.3. Struktur perulangan Repeat..Until

Struktur *Repeat..Until* digunakan untuk mengulang (*Repeat*) statemen-statemen atau blok statemensampai (*Until*) kondisi yang diseleksi di *Until* terpenuhi.

```

program ulang9;

uses crt;

var i : integer;

begin
clrscr;
i := 0;
Repeat
i := i + 1;
writeln(i);
until i = 5;
readln;
end.

```

Program 2.9 penggunaan dari perulangan Repeat ... Until

Bila program ini dijalankan, akan didapat hasil

- 1
- 2
- 3
- 4
- 5

Perbedaan antara struktur *Repeat..Until* dengan struktur *While-Do* adalah sebagai berikut :

1. Paling sedikit statemen-statemen di dalam perulangan *Repeat...Until* diproses sekali, karena seleksi kondisi ada pada statemen *Until* yang terletak dibawah.

Sedangkan pada Struktur *while-Do* paling sedikit dikerjakan nol kali, karena seleksi kondisi ada pada statemen *While* yang terletak diatas, sehingga kalau kondisi sudah tidak terpenuhi maka tidak akan masuk ke dalam lingkungan perulangannya.

Struktur Repeat..Until	Struktur While-Do
<pre>uses crt; var i : integer; begin i := 10; Repeat writeln(i); i := i + 1; Until i > 5; readln; end.</pre>	<pre>uses crt; var i : integer; begin clrscr; i := 10; while i < 5 do begin writeln(i); i := i + 1; end; readln; end.</pre>
Bila program dijalankan akan didapat hasil 10	Bila program dijalankan maka tidak akan didapatkan hasil.
Program 2.10 dan 2.11 prbandingan antara prulangan Repeat ..Until dengan perulangan While tentang jumlah perulangannya.	

2. Pada reapeat..until dapat tidak dipergunakan blok statemen(tidak diperlukan lagi *Begin* dan *End* untuk menunjukkan batas perulangan karena batas perulangannya sudah ditunjukkan oleh Repeat sampai dengan *Until* sebagai pengganti *Begin* dan *End* .

Contoh 12

```
program ulang12;

uses crt;

var celcius,fahrenheit : real;

begin
  clrscr;
  writeln('-----');
  writeln(' Celcius          Fahrenheit ');
  writeln('-----');
  celcius := 0;
  repeat
    fahrenheit := 1.8 * celcius + 32;
    writeln(Celcius:6:2, Fahrenheit:18:2);
    celcius := celcius + 0.5;
  until celcius > 5;
  writeln('-----');
  readln;
end.
```

Program 2.12 Perulangan Repeat ..Until tidak menggunakan blok statemen.

Bila program dijalankan, akan didapatkan hasil :

Celcius	Fahrenheit
0.00	32.00
0.50	32.90
1.00	33.80
1.50	34.70
2.00	35.60
2.50	36.50
3.00	37.40
3.50	38.30
4.00	39.20
4.50	40.10
5.00	41.00

3.1. Penilaian kompetensi

1. Buatlah program untuk menghitung rumus matematika jangkau dari masukkan adalah 0-255. Dan input data dari keyboard diulang sebanyak 3x. menggunakan For To Do
 - a) $Y1 = \sqrt{n} \times c + m^2 / r \times m$
 - b) $Y2 = Y1 / n \times 9 + y$

2. Buatlah Program untuk menampilkan pola-pola bintang seperti berikut

*	*	*
* *	* *	**
* * *	* * *	
* * * *		

Untuk n = 4

untuk n = 3

untuk n = 2

3. Tuliskanlah program untuk menampilkan semua bilangan ganjil yang kurang dari 100.
4. Tulislah program untuk mencari semua nilai x,y, dan z yang memenuhi persamaan linier berikut

$$12x - 9y + 8z = 0$$
 Dengan $1 \leq x, y, z \leq 100$.
5. [Konversi Bilangan Biner] tulislah program untuk mengkonversikan bilangan dalam format decimal menjadi bilangan dalam format biner. Masukkan dari program ini adalah bilangan bulat dalam format decimal. Keluaran dari program ini adalah string yang merupakan pengonversian masukan ke format biner.

Kunci Jawaban:

1.	<pre>program kompetensi1; uses crt; var Y1,Y2 : real; i : integer; n,m,r,c,y : byte; begin Writeln('-----'); for i := 1 to 3 do begin write('Masukkan data N = ');readln(n); write('Masukkan data C = ');readln(c); write('Masukkan data M = ');readln(m); write('Masukkan data R = ');readln(r); write('Masukkan data Y = ');readln(y); Y1 := (sqrt(n) * c + sqr(m))/(r*m); Y2 := (y1) / (n*9+y); writeln; writeln('Hasil dari Y1 adalah = ',Y1:2:2); writeln('Hasil dari Y2 adalah = ',Y2:2:2); end; readln; end.</pre>
2.	<pre>program kom3; uses crt; var N : byte; i,j : integer; begin clrscr; write('Masukkan N = ');readln(N); for i := 1 to N do begin for j := 1 to i do write('*'); writeln; end; readln; end.</pre>

3.	<pre> Program Komp2; uses crt; var i : integer; begin clrscr; i := 1; while (i < 100) do Begin writeln(i); i := i + 2; end; readln; end. </pre>
4.	<pre> program komp4; uses crt; var x,y,z : integer; const Max = 100; begin clrscr; for x := 1 to Max do for y := 1 to Max do for z := 1 to Max do begin if (12* x - 9 * y + 8 * z = 0) then writeln('x = ', x, ',y = ',y, ',z = ', z); end; readln; end. </pre>
5.	<pre> program komp5; uses crt; var x,sisa,j : integer; xb : string; begin clrscr; for j := 1 to 3 do begin </pre>

```
write('Masukkan x = ');readln(x);
write('x dalam format biner :');

xb := '';
sisas := 0;
repeat
  sisas := x mod 2;
  x := x div 2;
  xb := chr (sisas + $30) + xb;
until ( x = 0);

writeln (xb);

readln;
end;
end.
```

DAFTAR PUSTAKA

Anthony Pranata, 2000, *Algortima dan Pemograman*, J & J Learning Yogyakarta

Prof. Dr. Jogiyanto HM, MBA, Akt, 2005, *Turbo Pascal Versi 5.0 Teori dan Aplikasi Program Komputer Bahasa Turbo pascal Termasuk Database Toolbox Jilid 1*, Penerbit Andi Yogyakarta.