

**MODUL PRAKTIKUM
PEMROGRAMAN KOMPUTER I**



**TIM PENYUSUN:
TIM DOSEN**

**LABORATORIUM MATEMATIKA KOMPUTASI
JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS MULAWARMAN
SAMARINDA
2018**

PETUNJUK UMUM PRAKTIKUM

I. Ketentuan Umum

- a) Praktikum ini diperuntukkan bagi mahasiswa yang telah menempuh mata kuliah reguler sesuai dengan mata praktikum yang dipraktikkan dibuktikan dengan fotokopi KRS saat mendaftar praktikum.
- b) Peserta praktikum (praktikan) yang mengulang, diperlakukan sama dengan peserta reguler dan harus menempuh lagi mata kuliah tersebut.
- c) Praktikan harus mendaftar di laboratorium yang menjadi tempat praktikum mata kuliah tersebut.
- d) Praktikan mengikuti seluruh rangkaian kegiatan praktikum dan tata tertib selama pelaksanaan praktikum.

II. Pelaksanaan Praktikum

- a) Praktikum ini dilakukan sebanyak 10 (sepuluh) kali pertemuan dengan pertama adalah asistensi.
- b) Waktu yang disediakan untuk praktikum pada setiap pertemuan adalah 90 menit
- c) Praktikan yang terdaftar, akan dikelompokkan dengan setiap kelompok beranggotakan 15-20 praktikan dengan dipandu 2 Asisten Praktikum.
- d) Pelaksanaan praktikum dipandu oleh seorang Asisten Praktikum, yang bertindak sebagai Fasilitator, dan mereview setiap hasil presentasi pratikan.
- e) Praktikan diharuskan memahami konsep-konsep yang akan dibahas dalam praktikum, sesuai dengan materi yang ada.
- f) Semua fasilitas yang diperlukan oleh mahasiswa praktikan maupun yang diperlukan oleh Asisten Praktikum di dalam ruang praktikum seperti LCD, white board, spidol untuk white board, kertas, dan alat pendukung praktikum disediakan oleh Laboratorium.
- g) Setiap praktikan setelah mengikuti praktikum berhak mendapat nilai praktikum, dengan kategori nilai A-E.

III. Tata Tertib Praktikum

- a) Calon praktikan harus melakukan registrasi sendiri di Laboratorium dan tidak bisa diwakilkan
- b) Praktikan harus berpenampilan rapi dan bersih dengan kelengkapan baju berkerah, bersepatu tertutup, serta menggunakan atau memakai asesoris yang wajar

- c) Praktikan menghadiri pertemuan praktikum setidaknya sebanyak 8 (delapan) kali pertemuan
- d) Praktikan wajib membawa kartu kendali praktikum selama kegiatan praktikum
- e) Praktikan yang akan ijin tidak mengikuti praktikum wajib memberikan keterangan tertulis kepada asisten praktikum

DAFTAR ISI

HALAMAN JUDUL	i
PETUNJUK UMUM PRAKTIKUM	ii
DAFTAR ISI	iv
MODUL 1 PENGANTAR ALGORITMA DAN PEMROGRAMAN	
DASAR TEORI	1
LATIHAN	3
TUGAS	5
MODUL 2 PENGONDISIAN (<i>SELECTION</i>)	
DASAR TEORI	6
LATIHAN	7
TUGAS	9
MODUL 3 PERULANGAN (<i>REPETITION</i>)	
DASAR TEORI	10
LATIHAN	11
TUGAS	13
MODUL 4 <i>ARRAY</i> (LARIK)	
DASAR TEORI	14
LATIHAN	14
TUGAS	15
MODUL 5 <i>PROCEDURE</i> DAN <i>FUNCTION</i>	
DASAR TEORI	16
LATIHAN	17
TUGAS	18
MODUL 6 RECORD	
DASAR TEORI	19
LATIHAN	19
TUGAS	20
DAFTAR PUSTAKA	21

	LABORATORIUM MATEMATIKA KOMPUTASI JURUSAN MATEMATIKA	
	MATA KULIAH PRAKTIKUM : PEMROGRAMAN KOMPUTER I	
	MODUL : 1	MATERI : PENGANTAR ALGORITMA DAN PEMROGRAMAN

SUB CPMK MATA KULIAH

- Mampu menjelaskan konsep, pengertian, dan prinsip bahasa pemrograman.
- Mampu menjelaskan definisi pemrograman Pascal dan membuat suatu program berbahasa Pascal.

INDIKATOR

- Menjelaskan konsep bahasa pemrograman
- Menjelaskan prinsip bahasa pemrograman
- Menjelaskan definisi algoritma
- Memahami contoh-contoh program Pascal
- Menjelaskan definisi pemrograman Pascal
- Membuat struktur penulisan program dalam Pascal dengan benar
- Menggunakan tipe-tipe data dalam program Pascal
- Membuat program Pascal menggunakan sintaks standar, yakni *write*, *writeln*, *read*, dan *readln*.

DASAR TEORI

1.1 Pengantar Algoritma dan Pemrograman

Algoritma berisi langkah-langkah penyelesaian suatu proses/masalah. Setiap langkah didalam algoritma dinyatakan dalam sebuah pernyataan (*pernyataan*) atau istilah lainnya instruksi. Sebuah pernyataan berisi aksi (*action*) yang dilakukan.

Program adalah kumpulan instruksi/perintah yang diberikan kepada komputer untuk melaksanakan tugas tertentu. Sedangkan pemrograman adalah upaya untuk membuat kode-kode instruksi/perintah yang dapat dimengerti oleh komputer. Bahasa pemrograman adalah sarana yang digunakan untuk membuat kode-kode instruksi/perintah yang dapat dimengerti oleh komputer

1.2 Tipe Data, Variabel dan Operator

1.2.1 Tipe Data

Tabel 1.1 Tipe Data

Tipe Data Dalam Pascal	
Sederhana	Terstruktur
Integer Real Boolean Char	String Array Record Set File

Tabel 1.2 Interval tipe data nilai bulat dan nilai real

Tipe data dengan nilai bulat (integer)		Tipe data dengan nilai pecahan (Real)	
Tipe Data	Deskripsi (range variable)	Tipe Data	Deskripsi (range variable)
Byte	0 to 255	Real	$2,9 \times 10^{-38} \dots 1,7 \times 10^{38}$
Word	0 to 65535	Single	$1,5 \times 10^{-45} \dots 3,4 \times 10^{38}$
Integer	-32768 to 32767	Double	$5,0 \times 10^{-324} \dots 1,7 \times 10^{308}$
Longint	-2147483648 to 2147483647	Extended	$3,4 \times 10^{-4932} \dots 1,11 \times 10^{4932}$
Shortint	-128 to 127	Comp	$-2^{63} + 1 \dots 2^{63} - 1$

1.2.2 Variabel

Variabel adalah besaran yang nilainya dapat berubah-ubah. Variabel dapat digunakan dalam tubuh program bila telah didefinisikan terlebih dahulu pada bagian deklarasi

1. Nama variabel bebas (tergantung pemrogram) namu harus mengikuti aturan sebagai berikut :
 - Bukan merupakan kata perintah khusus (reserved word) dalam bahasa pascal
 - Tidak menggunakan spasi (penghubung yang digunakan adalah garis bawah [_])
 - Tidak menggunakan simbol-simbol khusus seperti : +,=,-,*,/, \$ dll
 - Tidak didahului dengan angka
 - Nama variabel jangan terlalu panjang dan sebaiknya mudah diingat

2. Format Definisi Variabel

```
Var Nama_Variabel : Tipe_Data;
Atau
Var
    Nama_Variabel : Tipe_Data;
Contoh :
Var Nama : String;
Atau
Var
    Nama_Mahasiswa : String;
```

1.2.3 Operator

1. Operator Aritmatika

Tabel 1.3 Operator Aritmatika

Operator	Operasi	Tipe Operand	Tipe Hasil Operasi
+	Penjumlahan	Integer, real	Integer, real
-	Pengurangan	Integer, real	Integer, real
*	Perkalian	Integer, real	Integer, real
/	Pembagian	Integer, real	Integer, real
div	Pembagian	Integer, integer	Integer
mod	Sisa pembagian	Integer, integer	Integer

2. Operator Relasional

Tabel 1.4 Operator Relasional

Operator	Operasi
=	Sama dengan
<>	Tidak sama dengan
<	Lebih kecil dari
>	Lebih besar dari
<=	Lebih kecil atau =

>=	Lebih besar atau =
----	--------------------

LATIHAN

1. Dibawah ini merupakan salah satu program sederhana yang mengeluarkan sebuah pernyataan.

```
Program Contoh1;
Uses wincrt;
begin
  Writeln('KAMI ANAK STATISTIKA 2015!');
end.
```

2. Untuk membuat variabel di dalam pascal, kita harus mendeklarasikannya sebelum *main program*. Setiap variabel juga memiliki tipe data tertentu, dan sepanjang kode program, variabel tersebut hanya dapat diubah nilainya asalkan masih dalam tipe yang sama. Untuk memperjelas pernyataan tersebut, dapat dilihat dari contoh dibawah.

```
Program Contoh2;
Uses Wincrt;
Var
  bil1,bil2,L:integer;
Begin
  Writeln('PROGRAM PENJUMLAHAN');
  Writeln;
  Write('Masukkan Nilai Pertama = ');Readln(bil1);
  Write('Masukkan Nilai Kedua   = ');Readln(bil2);
  L:=bil1+bil2;
  Writeln('Jadi hasil penjumlahannya adalah ->
',bil1,'+',bil2,'=',L);
end.
```

3. Selanjutnya untuk menampilkan karakter yang telah disediakan, dapat menggunakan perintah **write** dan **writeln**. Perbedaan untuk kedua perintah ini hampir sama dengan konsep **read** dan **readln**. Untuk lebih jelasnya, dapat dipraktikkan pada latihan untuk memasukkan identitas karyawan swasta di Samarinda berikut ini.

```
Program latihan1_1;
uses wincrt;
var
  nama,KTP,npwp,alamat,Pekerjaan:string;
  usia:string[2];
```



```

begin
clrscr;
writeln('-----');
writeln('IDENTITAS DIRI');
writeln('-----');
write('nama anda: ');readln(nama);
write('usia anda: ');readln(usia);
write('alamat anda: ');readln(alamat);
write('no. KTP anda: ');readln(ktp);
write('no. NPWP anda: ');readln(npwp);
write('pekerjaan anda: ');readln(pekerjaan);
end.

```

Perhatikan bahwa pendeklarasian variabel usia dapat menggunakan perintah **usia:string[2];**. Hal ini berarti variabel usia bertipe *string* dengan batas maksimal 2 karakter. Secara umum, untuk mendeklarasikan variabel usia adalah menggunakan karakter bertipe numerik.

TUGAS

1. Buatlah program yang menampilkan kata sebagai berikut:

```

Selamat      Datang      di      Laboratorium      Matematika
komputasi
Jagalah Kebersihan dan Kenyamanan Sekitar
Rapikan Tempat Duduk Setelah Praktikum Selesai

```

2. Buatlah program untuk menghitung luas lingkaran!
3. Buatlah program untuk menghitung konversi dari celcius ke reamur!

	LABORATORIUM MATEMATIKA KOMPUTASI JURUSAN MATEMATIKA	
	MATA KULIAH PRAKTIKUM : PEMROGRAMAN KOMPUTER I	
	MODUL : 2	MATERI : PENGONDISIAN (<i>SELECTION</i>)

SUB CPMK MATA KULIAH

- Mampu menjelaskan sintaks kondisi if, dan membuat suatu program Pascal menggunakan sintaks kondisi if.
- Mampu menjelaskan sintaks seleksi case, dan membuat suatu program Pascal menggunakan sintaks kondisi case.

INDIKATOR

- Menjelaskan penggunaan sintaks untuk kondisi if...then
- Menjelaskan penggunaan sintaks untuk kondisi if..then else...
- Membuat program Pascal menggunakan sintaks kondisi if.
- Menjelaskan penggunaan sintaks untuk seleksi Case...Of
- Membuat program Pascal menggunakan sintaks seleksi Case...Of

DASAR TEORI

2.1 Algoritma Pemilihan (*Selection*)

Blok pemilihan merupakan blok kontrol yang digunakan untuk memilih pernyataan atau aksi-aksi yang akan dilakukan, dimana pemilihan tersebut didasarkan atas nilai dari kondisi-kondisi tertentu. Suatu aksi akan dikerjakan atau dieksekusi oleh program apabila kondisi yang didefinisikan untuk aksi tersebut bernilai benar (*true*). Sebaliknya, bila kondisi tidak terpenuhi atau salah (*false*) maka program akan melakukan aksi lain (jika ada) atau langsung keluar dari blok pemilihan. Dalam bahasa pascal, blok pemilihan dapat didefinisikan dengan dua cara, yaitu dengan menggunakan pernyataan *if* dan *case*.

2.1.1 Pernyataan *if*

Pernyataan *if* akan diikuti oleh ekspresi (sebagai kondisi yang akan diperiksa) dan selalu berpasangan dengan kata kunci *then*. Apabila pernyataan yang akan dilakukan hanya satu maka kita tidak perlu menuliskan blok *begin..end*. Namun apabila pernyataan lebih dari satu maka blok *begin..end* harus dituliskan. Pernyataan IF-THEN-ELSE digunakan jika terdapat lebih dari satu pernyataan, pernyataan pertama akan dieksekusi apabila kondisi bernilai *true* dan sebaliknya pernyataan kedua akan dieksekusi apabila kondisi bernilai *false*.

Sintaks umum dari if – then dengan satu pernyataan:

```
if <kondisi> then <pernyataan>;
```

Sintaks umum dari if – then – else dengan dua pernyataan:

```
if <kondisi> then <pernyataan 1>
else <pernyataan 2>;
```

2.1.2 Pernyataan Case

Pernyataan *case* berisi ungkapan (pemilih) dan sederetan pernyataan, yang masing-masing diawali dengan satu atau lebih konstan (disebut konstanta *case*) atau dengan kata *else*. Pernyataan *case* dapat berbentuk struktur *Case-Of* atau *Case-Of...Else*. *Case of* adalah bahasa pemrograman agar output pada pemrograman pascal mengikuti sesuai dengan input. Bentuk umum sintaks dari *case of*:

```
Case<variabel> of
  <kondisi1>:<pernyataan1>;
  <kondisi2>:<pernyataan2>;
  <kondisi3>:<pernyataan3>;
end;
```

LATIHAN

IF THEN ELSE

Dengan perintah ***if..then..else***, pernyataan-pernyataan akan dikerjakan jika kondisi yang diberikan terpenuhi. Untuk lebih jelasnya, mari ikuti dua latihan berikut.

```
program latihan2_1;
uses wincrt;
var
  nama,nim:string[25];
  nilai:real;
begin
  writeln('    > > >Selamat Datang Di< < <');
  writeln('Program Nilai Komputer Dasar');
  writeln;
  write('Nama Anda: ');readln(nama);
  write('NIM Anda : ');readln(nim);
  write('Nilai    : ');readln(nilai);
  if (nilai<=100) and (nilai>=80) then
    writeln('Hebat, anda mendapat nilai A')
  else
    if (nilai<80) and (nilai>=70) then
      writeln('Selamat, anda mendapat nilai B')
    else
      if (nilai<70) and (nilai>=55) then
        writeln('Cukup bagus, anda mendapat nilai C')
      else
```

```

        if (nilai<55) and (nilai>=40) then
            writeln('Maaf, anda mendapat nilai D')
        if (nilai<40) and (nilai>=0) then
            writeln('Anda tidak lulus, anda mendapat
nilai E');
end.

```

Selain contoh di atas dapat lihat contoh lainnya sebagai berikut:

```

program latihan2_2;
Uses wincrt;
Var
    Nil1, Nil2 : Integer;
    NilRata    : Real;
Begin
    ClrScr;
    Write('Masukan Nilai 1 : ');
    Readln(Nil1);
    Write('Masukan Nilai 2 : ');
    Readln(Nil2);
    NilRata := (Nil1 + Nil2)/2;
    Writeln('Nilai Rata-Rata = ', NilRata:5:2);
    if NilRata >= 60 then
        Writeln('Hasilnya = Lulus')
    else
        Writeln('Hasilnya = Gagal');
    Readln;
end.

```

CASE OF

Berbeda dengan perintah pengkondisian ***if..then..else***, yang memanfaatkan logika *true* dan *false* (dua kemungkinan); perintah ***case of*** dapat digunakan untuk memilih dengan kemungkinan yang lebih dari dua.

```

Program latihan2_3;
uses wincrt;
var
    pilih:char;

begin
    writeln('Selamat Datang di Warung Polah Etam
Sukses');
    writeln('d^_^');

```

```

writeln;
writeln('> > >M E N U< < <');
writeln('A. Jukut Tunu');
writeln('B. Soto Banjar');
writeln('C. Nasi Kuning Biasa');
writeln('D. Nasi Kuning Spesial');
writeln('E. Nasi Goreng Biasa');
writeln('F. Nasi Goreng Spesial');
writeln('G. Keluar');
write('Silahkan pilih [A-G]: ');readln(pilih);
clrscr;

    case pilih of
'A':writeln('Anda memesan Jukut Tunu, harga Rp
20.000');
'B':writeln('Anda memesan Soto Banjar, harga Rp
15.000');
'C':writeln('Anda memesan Nasi Kuning Biasa, harga Rp
10.000');
'D':writeln('Anda memesan Nasi Kuning Spesial, harga
Rp 12.000');
'E':writeln('Anda memesan Nasi Goreng Biasa, harga
10.000');
'F':writeln('Anda memesan Nasi Goreng Spesial, harga
Rp 12.000');
'G':exit;
    end;
end.

```

TUGAS

1. Buatlah program untuk mencari nilai maksimum dan minimum dari 2 buah bilangan bulat yang berbeda!
2. Buatlah program dari 3 buah bilangan bulat yang berbeda dan menampilkan bilangan terkecil diantara 3 bilangan tersebut!
3. Buat program yang dapat menentukan besarnya nilai FPB (Faktor Persekutuan Terbesar) diantara dua bilangan bulat yang berbeda!

	LABORATORIUM MATEMATIKA KOMPUTASI JURUSAN MATEMATIKA	
	MATA KULIAH PRAKTIKUM : PEMROGRAMAN KOMPUTER I	
	MODUL : 3	MATERI : PERULANGAN (<i>REPETITION</i>)

SUB CPMK MATA KULIAH

- Mampu menjelaskan sintaks perulangan for, dan membuat suatu program Pascal menggunakan sintaks perulangan for.
- Mampu menjelaskan sintaks perulangan while, dan membuat suatu program Pascal menggunakan sintaks perulangan while.
- Mampu menjelaskan sintaks perulangan repeat, dan membuat suatu program Pascal menggunakan sintaks perulangan repeat, serta membuat suatu program Pascal bersarang.

INDIKATOR

- Menjelaskan penggunaan sintaks untuk perulangan for...to...do
- Membuat program Pascal sederhana menggunakan sintaks perulangan for...to...do
- Membuat program Pascal sederhana menggunakan sintaks perulangan for...to...do dalam studi kasus *sorting* (pengurutan)
- Menjelaskan penggunaan sintaks untuk perulangan while...do
- Membuat program Pascal menggunakan sintaks perulangan while...do
- Menjelaskan penggunaan sintaks untuk perulangan repeat...until...
- Membuat program Pascal menggunakan sintaks perulangan repeat...until...
- Membuat program Pascal bersarang menggunakan sintaks perulangan (for, while, dan repeat).

DASAR TEORI

3.1 Algoritma Pengulangan (*Repetition*)

Perulangan adalah suatu pernyataan yang digunakan untuk mengerjakan setiap atau beberapa pernyataan secara berulang-ulang. Di dalam Pascal, dikenal tiga macam perulangan, yaitu dengan menggunakan pernyataan *For*, *While-Do* dan *Repeat-Until*.

3.1.1 Perulangan *For*

1. Perulangan Positif

Perulangan positif adalah perulangan dengan perhitungan dari kecil ke besar atau pertambahan positif. Bentuk umum perulangan for positif adalah:

```
for <variabel kontrol>:=<nilai awal> to <nilai akhir> do< pernyataan>
```

2. Perulangan Negatif

Perulangan negatif adalah perulangan dengan perhitungan dari besar ke kecil atau pertambahan negatif. Bentuk umum perulangan *for* negatif adalah:

```
for <variabel kontrol>:=<nilai awal> downto <nilai akhir> do <pernyataan>
```

3.1.2 Perulangan *While-Do*

Perulangan *while-do* digunakan untuk melakukan proses perulangan suatu pernyataan atau blok pernyataan terus menerus selama kondisi ungkapan logika pada *while* masih bernilai logika benar atau terpenuhi. Bentuk umum dari perulangan *while-do* adalah sebagai berikut:

```
while<kondisi ungkapan> do <pernyataan>
```

3.1.3 Perulangan *Repeat-Until*

Perulangan *repeat-until* digunakan untuk mengulang pernyataan-pernyataan atau blok pernyataan sampai kondisi yang diseleksi di *until* tidak terpenuhi. Bentuk umumnya adalah:

```
repeat <pernyataan> until <kondisi>
```

LATIHAN

FOR

Pada perintah **for**, perulangan dilakukan sebanyak jumlah yang telah ditentukan.

```
program latihan3_1;
uses wincrt;
var
  i : integer;

begin
  for i := 1 to 5 do
    writeln ('Mari Belajar Komputer Dasar');
    readln;
  end.
```

Selain perulangan seperti pada latihan3_1, perintah **for** juga dapat digunakan untuk perulangan negatif; yaitu penghitung (*counter*) dari besar ke kecil atau pertambahan indeksinya negatif. Perhatikan penggunaan indeks pada latihan3_2 berikut.

```
program latihan3_2;
uses wincrt;
var
  i : integer;
```

```
begin
  for i := 10 downto 1 do
    write (i:3);
  end.
```

WHILE DO

Selanjutnya adalah perulangan **while..do**. Berbeda dengan perintah **for** dengan jumlah perulangan yang telah ditentukan. Pada perintah **while..do**, selama kondisi ungkapan logika masih benar maka proses perulangan statemen-statemen dilakukan terus menerus. Lebih jelasnya perhatikan latihan membuat tabel perhitungan menggunakan perintah **while..do** berikut.

```
program latihan3_3;
uses wincrt;
var
  bawah,atas:integer;
  kuadrat:real;
begin
  writeln('Menghitung nilai Xi dan Xi^2 dalam suatu
interval');
  writeln;
  write('masukkan batas bawah: ');readln(bawah);
  write('masukkan batas atas : ');readln(atas);
  clrscr;
  writeln(' --- -----');
  writeln('| x | x^2 |');
  writeln(' --- -----');
  while (bawah<=atas) do
    begin
      kuadrat:=sqr(bawah);
      writeln('|',bawah:2,' |',kuadrat:4:0,' |');
      bawah:=bawah+1;
    end;
  writeln(' --- -----');
end.
```

REPEAT UNTIL

Selain perintah **for..do** dan **while..do**, bahasa pemrograman pascal juga mengenal perintah **repeat..until**. Berikut adalah latihan perulangan

repeat..until dalam menghitung nilai rata-rata untuk data yang akan dimasukkan.

```

program latihan3_4;
uses wincrt;
var
    rata2:real; i,n,a,jumlah:integer;
begin
    i:=1;
    writeln('Program Menghitung Rata-Rata');
    write('ada berapa banyak Data? ');readln(n);
    writeln;
    repeat
        begin
            write('masukkan data ke-',i,': ');readln(a);
            jumlah:=jumlah+a;
            i:=i+1;
        end;
    until (i=n+1);
    rata2:=jumlah/n;
    writeln;
    writeln('rata-rata data tersebut adalah:
',rata2:0:2);
end.

```

TUGAS

1. Buatlah program menghitung deret kuadrat dengan menggunakan for down to!
2. Buatlah program yang menerima sebuah bilangan bulat N, lalu menghitung hasil penjumlahan $1 + 2 + \dots + N$!
3. Buatlah program yang menghasilkan nilai variansi!

	LABORATORIUM MATEMATIKA KOMPUTASI JURUSAN MATEMATIKA	
	MATA KULIAH PRAKTIKUM : PEMROGRAMAN KOMPUTER I	
	MODUL : 4	MATERI : ARRAY

SUB CPMK MATA KULIAH

- Mampu menjelaskan sintaks *array* 1 dimensi, dan membuat suatu program Pascal menggunakan sintaks *array* 1 dimensi.
- Mampu menjelaskan sintaks *array* 2 dimensi, dan membuat suatu program Pascal menggunakan sintaks *array* 2 dimensi.

INDIKATOR

- Menjelaskan definisi dan penggunaan sintaks untuk *array* 1 dimensi
- Membuat program Pascal menggunakan sintaks *array* 1 dimensi
- Menjelaskan definisi dan penggunaan sintaks untuk *array* 2 dimensi
- Membuat program Pascal menggunakan sintaks *array* 2 dimensi.

DASAR TEORI

4.1 Array

Menurut definisinya, *array* atau larik adalah sebuah variabel yang dapat menyimpan lebih dari satu nilai sejenis (memiliki tipe data yang sama). Hal ini tentu berbeda dengan variabel yang biasa hanya menampung satu buah nilai. Setiap nilai yang disimpan dalam *array* disebut Elemen *Array*, sedangkan nilai urut yang digunakan untuk mengakses elemennya disebut dengan indeks *array*.

Sintaks Umum Array

```
NamaArray:array[IndeksAwal..IndeksAkhir)of tipe_data;
```

Sintaks Array Dua Dimensi

```
NamaArray: array[1..BanyakBaris, 1..BanyakKolom];
```

Sintaks Array Tiga Dimensi

```
NamaArray:array[1..xMaks, 1..yMaks, 1..zMaks] of  
tipe_data;
```

LATIHAN

Array digunakan sebagai susunan yang terdiri dari sejumlah komponen yang mempunyai tipe yang sama. Untuk lebih jelasnya, ikuti latihan untuk merekap data mahasiswa FMIPA UNMUL berikut.

```
program latihan4;  
uses wincrt;  
type  
    data=array[1..50] of string;  
var
```

```

nama,nim,prodi:data;
n,i:integer;
begin
    gotoxy(20,1);write('    <><>Data    Mahasiswa
FMIPA<><>');

gotoxy(20,2);writeln('<*><*><*><*><*><*><*>');
writeln;
write('ada berapa Mahasiswa? ');readln(n);
clrscr;
for i:=1 to n do
begin
    writeln('Mahasiswa ke-',i);
    write('Nama    : ');readln(nama[i]);
    write('NIM     : ');readln(nim[i]);
    write('Jurusan: ');readln(prodi[i]);
    clrscr;
end;
    gotoxy(20,1);write('    <><>Data    Mahasiswa
FMIPA<><>');

gotoxy(20,2);write('<*><*><*><*><*><*><*>');
writeln;
for i:=1 to n do
begin
    writeln('Nama    : ',nama[i]);
    writeln('NIM     : ',nim[i]);
    writeln('Jurusan: ',prodi[i]);
    writeln('-----');
end;
end.

```

TUGAS

1. Buatlah program larik untuk menghitung gaji seorang pegawai!
2. Buatlah program larik untuk menyimpan data!
3. Buatlah program yang menyimpan data ke larik kemudian menentukan nilai rata-rata dan standar deviasi dari seluruh data tersebut!

	LABORATORIUM MATEMATIKA KOMPUTASI JURUSAN MATEMATIKA	
	MATA KULIAH PRAKTIKUM : PEMROGRAMAN KOMPUTER I	
	MODUL : 5	MATERI : PROCEDURE DAN FUNCTION

SUB CPMK MATA KULIAH

- Mampu menjelaskan sintaks penulisan *procedure*, dan membuat suatu program Pascal menggunakan sintaks penulisan *procedure*.
- Mampu menjelaskan sintaks penulisan *function*, dan membuat suatu program Pascal menggunakan sintaks penulisan *function*.

INDIKATOR

- Menjelaskan definisi dan penggunaan sintaks penulisan *procedure*
- Membuat program Pascal menggunakan sintaks penulisan *procedure*.
- Menjelaskan definisi dan penggunaan sintaks penulisan *function*
- Membuat program Pascal menggunakan sintaks penulisan *function*.

DASAR TEORI

5.1 Procedure

Procedure atau prosedur merupakan suatu rutin yang melakukan proses tertentu tanpa adanya pengembalian nilai. Prosedur menyediakan suatu metode pengisolasian bagian program secara terpisah yang dapat dipanggil atau diaktivasi dari bagian-bagian manapun di dalam program. Suatu prosedur biasanya memiliki satu atau lebih parameter (Raharjo, 2005).

Sintaks umum prosedur:

```
Procedure nama_prosedur(parameter:tipe_data);
Begin
<pernyataan>;
End;
```

5.2 Function

Definisi *function* atau fungsi sebenarnya sama dengan sebuah prosedur. Perbedaannya, pada fungsi terdapat pengembalian nilai, sehingga pada saat pemanggilan, fungsi dapat langsung digunakan untuk mengisikan sebuah ekspresi. Berbeda dengan prosedur yang didefinisikan dengan kata kunci *procedure*, fungsi didefinisikan dengan kata kunci *function* (Raharjo, 2005).

Sintaks umum fungsi:

```
function Nama_Fungsi(parameter:tipe_data):tipe_data;
begin
```

```
<pernyataan>;End;
```

LATIHAN

PROCEDURE

Perintah **procedure** harus dideklarasikan terlebih dahulu. Berikut adalah latihan menghitung luas dan keliling segitiga.

```
program latihan5_1;
uses wincrt;
var
    l,k:real; a,t:byte;

procedure hitung(alas,tinggi:byte);
begin
    l:=(alas*tinggi);
    k:=1/2*(alas*tinggi);
    writeln('luas segitiga adalah= ',l:3:0);
    writeln('keliling segitiga adalah= ',k:3:0);
end;

begin
    write('masukkan nilai alas= ');readln(a);
    write('masukkan nilai tinggi= ');readln(t);
    hitung(a,t);
end.
```

FUNCTION

Sepertihalnya **procedure**, perintah **function** dideklarasikan terlebih dahulu. Namun perintah **function** harus dideklarasikan bersama type datanya.

```
program latihan5_2;
uses wincrt;
Function Hitung(Var A,B : integer): integer;
begin
    Hitung := A + B;
End;
Var X,Y : integer;
begin
    Write('Nilai X ? ');
    Readln(X);
    Write('Nilai Y ? ');
```

```
Readln(Y);  
Writeln;  
Writeln(X, ' + ', Y, ' = ', Hitung(X, Y)); End.
```

TUGAS

1. Buatlah program untuk menghitung luas sebuah bangun!
2. Buatlah program untuk menukar 2 buah huruf abjad!
3. Buatlah program untuk mencari nilai pertambahan, pengurangan, perkalian dan pembagian!

	LABORATORIUM MATEMATIKA KOMPUTASI JURUSAN MATEMATIKA	
	MATA KULIAH PRAKTIKUM : PEMROGRAMAN KOMPUTER I	
	MODUL : 6	MATERI : RECORD

SUB CPMK MATA KULIAH

- Mampu menjelaskan sintaks penulisan *record*, dan membuat suatu program Pascal menggunakan sintaks penulisan *record*.
- Mampu menjelaskan sintaks penulisan *record*, dan membuat suatu program Pascal menggunakan sintaks penulisan *record*.

INDIKATOR

- Menjelaskan definisi dan penggunaan sintaks penulisan *record*
- Membuat program Pascal menggunakan sintaks penulisan *record*.
- Membuat program Pascal menggunakan sintaks penulisan *record* dengan pernyataan *with...do*
- Membuat program Pascal dalam kehidupan sehari-hari menggunakan sintaks penulisan *record*

DASAR TEORI

6.1 Record

Record merupakan salah satu tipe struktur dalam bahasa Pascal yang mengumpulkan hubungan beberapa item data yang berbeda tipe datanya. Masing-masing item pada record disebut sebuah bidang.

Sintaks umum *record*:

```
Type Nama_record = RECORD
Bidang_1:tipe_bidang_1;
Bidang_2:tipe_bidang_2;
..
Bidang_n:tipe_bidang_n;
End;
```

LATIHAN

Dalam suatu toko dibuat informasi mengenai biodata karyawannya, dengan menggunakan *record* dibuat program sebagai berikut:

```
program latihan6;
uses wincrt;
TYPE
Karyawanrec=RECORD
Nama      :string[25];
Alamat    :string[40];
Hp        :string[12];
Gaji      :real;
```

```

Status :string[15];
End;
Var
    Karyawan:karyawanrec;
Begin
    Karyawanrec.nama:='Santo Christyadi';
    Karyawanrec.alamat:='Bukuan Palaran';
    Karyawanrec.hp:='081351710095';
    Karyawanrec.gaji:=3.67;
    Karyawan.status:='belum menikah'
    Writeln('Nama Karyawan   : ',karyawanrec.nama);
    Writeln('Alamat           : ',karyawanrec.alamat);
    Writeln('Nomor Handphone: ',karyawanrec.hp);
    Writeln('Gaji Sebulan    : Rp',karyawanrec.gaji,'
Juta);
    Writeln('Status Perkawinan: ',karyawanrec.status);
    Readln;
end.

```

TUGAS

Buatlah program record dengan menggunakan bidang-bidang sebagai berikut!

- Data Karyawan: Nama, ID, Ntework Domai, Email, dan Kantor
- Data Bayi Baru Lahir: Nama, Nama Ayah, Nama Ibu, Berat, Tempat Lahir, dan tanggal lahir
- Jadwal penerbangan: Nama Maskapai, Nomor Penerbangan, Jam Keberangkatan, Kota Berangkat, Jam Kedatangan, dan Kota Kedatangan.

DAFTAR PUSTAKA

- Insap Santosa, P. 2000. *Struktur Data Menggunakan Turbo Pascal 6.0*. Yogyakarta : Andi Offset
- Raharjo, Budi. 2005. *Teknik Pemrograman Pascal*. Bandung: Informatika Bandung.
- Suprpto. 2008. *Bahasa Pemrograman*. Jakarta: Erlangga.