

# **BAB I ALGORITMA DAN PEMROGRAMAN**

## **1.1. DEFINISI ALGORITMA**

**“Algoritma adalah urutan langkah-langkah logis penyelesaian masalah yang disusun secara sistematis logis”.**

### **A. Algoritma Merupakan Jantung Ilmu Informatika**

Algoritma adalah jantung ilmu computer atau imformatika. Banyak cabang ilmu computer yang diacu dalam terminology algoritma. Namun, jangan beranggapan algoritma selalu identik dengan ilmu computer saja. Dalam kehidupan sehari-hari banyak terdapat proses yang dinyatakan dalam suatu algoritma. Cara-cara membuat kue atau masakan yang dinyatakan dalam suatu resep juga dapat disebut algoritma. Pada setiap resep selalu ada urutan langkah-langkah membuat masakan. Bila langkah-langkah tidak logis, tidak dapat dihasilkan masakan yang diinginkan. Ibu-ibu yang mencoba suatu resep masakan akan membaca satu persatu langkah-langkah pembuatannya lalu ia mengerjakan proses sesuai yang ia baca. Secara umum, pihak (benda) yang mengerjakan proses disebut pemroses (processor). Pemroses tersebut dapat berupa manusia, computer, robot atau alat-alat elektronik lainnya. Pemroses melakukan suatu proses dengan melaksanakan atau “mengeksekusi” algoritma yang menjabarkan proses tersebut.

Melaksanakan Algoritma berarti mengerjakan langkah-langkah di dalam Algoritma tersebut. Pemroses mengerjakan proses sesuai dengan algoritma yang diberikan kepadanya. Juru masak membuat kue berdasarkan resep yang diberikan kepadanya, pianis memainkan lagu berdasarkan papan not balok. Karena itu suatu Algoritma dinyatakan dalam bentuk yang dimengerti oleh pemroses. Jadi suatu pemroses harus :

1. Mengerti setiap langkah dalam algoritma
2. Mengerjakan operasi yang bersesuaian dengan langkah tersebut.

### **B. Belajar Memprogram dan Belajar Bahasa Pemrograman**

Belajar memprogram tidak sama dengan belajar bahasa pemrograman. Belajar memprogram adalah belajar tentang metodologi pemecahan masalah, kemudian menuangkannya dalam suatu notasi tertentu yang mudah dibaca dan dipahami. Sedangkan belajar bahasa pemrograman berarti belajar memakai suatu bahasa aturan-aturan tata bahasanya, instruksi-instruksinya, tata cara pengoperasian compilernya, dan memanfaatkan instruksi-instruksi tersebut untuk membuat program yang ditulis hanya dalam bahasa itu saja.

### **C. Langkah Sistematis dalam Membuat Program**

Dalam menyusun sebuah program computer, And harus melakukan langkah-langkah sistematis berikut ini.

#### **1. Mengidentifikasi Masalah**

Yang dimaksud dengan mengidentifikasi masalah adalah mengenal dan memahami masalah

#### **2. Merumuskan Pemecahan Masalah**

Dalam merumuskan pemecahan masalah, perlu mengenal algoritma, bentuk algoritma ada beberapa bentuk yaitu algoritma dalam bahasa sehari-hari, pseudocode, dan algoritma dalam bentuk diagram alir

#### **3. Membuat/ Menulis Program**

Untuk menulis program, diperlukan pengetahuan minimal sebuah bahasa pemrograman. Dalam langkah ini diterapkan pemahaman logika tentang permasalahan dan solusi anda ke dalam bentuk program computer.

#### **4. Evaluasi**

Langkah evaluasi ini digunakan untuk memeriksa kembali logika pemrograman pada bahasa program yang ditulis, jika terdapat kesalahan, kembali ke tahap penulisan program. Dalam langkah ini

tercakup proses uji coba kecocokan output programnya yang dihasilkan dengan algoritma yang telah disusun.

### Contoh Soal (Permasalahan)

Tentukan Algoritma dari soal-soal berikut

1. Menghitung luas persegi panjang dengan data yang diketahui berupa panjang dan lebarnya.
2. Menghitung luas lingkaran dengan data yang diketahui berupa diameter

Jawab

#### 1. Luas Persegi Panjang

##### a. **Identifikasi Masalah**

Point yang harus ada Panjang, Lebar dan Luas

##### **Merumuskan Masalah**

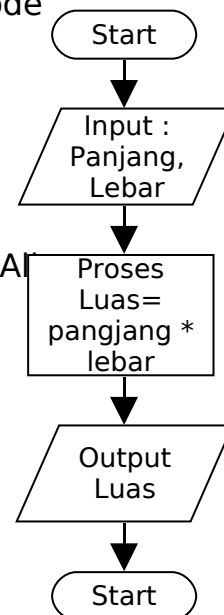
##### 1) Algoritma dalam bahasa sehari-hari

- Mulai
- Tentukan panjangnya
- Tentukan lebarnya
- Hitung Luas L dengan rumus  $L = \text{panjang} \times \text{lebar}$
- Tampilkan nilai L
- Selesai

##### 2) Algoritma dalam bentuk Pseudocode

- Start
- Input Panjang
- Input Lebar
- $L = \text{panjang} * \text{lebar}$
- Output L
- End.

##### 3) Algoritma dalam Bentuk Diagram Alir



#### 2. Luas Lingkaran (Dik. Diameter)

##### a. **Identifikasi Masalah**

Point yang harus ada Diameter, Jari-jari, Luas lingkaran

##### **Merumuskan Masalah**

##### 4) Algoritma dalam bahasa sehari-hari

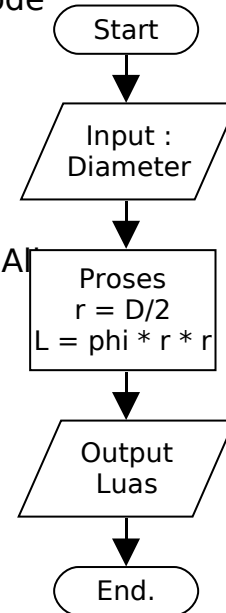
- Mulai

- Tentukan Diameter
- Hitung jari-jari  $r$  dengan rumus  $= D/2$
- Hitung Luas  $L$  dengan rumus  $L = \pi * r * r$
- Tampilkan nilai  $L$
- Selesai

5) Algoritma dalam bentuk Pseudocode

- Start
- Input Diameter
- $r = D/2$
- $L = \pi * r * r$
- Output  $L$
- End.

6) Algoritma dalam Bentuk Diagram Alir



### Latihan Soal

Buatlah sebuah urutan langkah (Algoritma sederhana dalam bahasa sehari-hari) untuk soal berikut

- a. Menghitung luas sebuah segitiga dengan input data berupa panjang alas, dan tinggi segitiga tersebut.
- b. Menghitung luas lingkaran dengan input data berupa keliling lingkaran tersebut.
- c. Menghitung laba jika diketahui harga penjualan dan harga pembelian
- d. Menghitung diskon jika diketahui harga penjualan sebelum discout dan harga penjualan setelah discout.

### 1.2. Struktur Dasar Algoritma

Algoritma berisi langkah-langkah penyelesaian suatu masalah. Langkah-langkah tersebut dapat berupa runtunan aksi, pemilihan aksi, dan pengulangan aksi. Ketiga jenis langkah tersebut membentuk konstruksi suatu algoritma. Jadi, sebuah algoritma dapat dibangun dari tiga buah struktur dasar, yaitu :

#### 1. Runtunan

Sebuah runtunan terdiri dari satu atau lebih instruksi. Tiap instruksi dikerjakan secara berurutan sesuai dengan urutan penulisannya. Yakni sebuah instruksi dilaksanakan setelah instruksi sebelumnya selesai dilaksanakan. Urutan instruksi menentukan keadaan akhir algoritma. Bila urutannya diubah, maka hasil akhirnya mungkin juga berubah. Perhatikan urutan instruksi yang dilambangkan dengan A1, A2, A3, A4, A5 berikut

A1

A2

A3

A4

A5

Mula-mula pemroses melaksanakan instruksi A1, instruksi A2 dilaksanakan setelah instruksi A1 selesai. Selanjutnya instruksi A3 dilaksanakan setelah instruksi A2 selesai. Demikian seterusnya

sampai instruksi terakhir A5 dilaksanakan. Setelah instruksi A5 dilaksanakan, algoritma berhenti.

## 2. Pemilihan

Adakalanya sebuah instruksi dikerjakan jika kondisi tertentu dipenuhi. Tinjau kembali algoritma pengurutan data.

Pemilihan dapat ditulis dalam struktur umum :

### **If kondisi then aksi**

Dalam bahasa Indonesia, **if** berarti “jika” dan **then** artinya “maka”. **Kondisi** adalah persyaratan yang dapat bernilai benar atau salah. **Aksi** hanya dilaksanakan apabila kondisi bernilai benar. Sebaliknya, apabila kondisi bernilai salah, maka aksi tidak dilaksanakan. Perhatikan bahwa kata yang digarisbawahi, **if dan then** merupakan kata kunci (keywords) untuk struktur pemilihan ini.

Dalam kehidupan sehari-hari, kita sering menuliskan pelaksanaan tindakan bila suatu persyaratan dipenuhi, misalnya :

**If** Amir memperoleh juara kelas **then**  
Ayah akan membelikannya sepeda

**If** jalan Dago macet **then**  
Ambil alternative jalan Dipati Ukur

**If** mobilmu rusak **then**  
Pakai saja sepeda motorku

Struktur pemilihan **if-then** hanya memberikan satu pilihan aksi bila kondisi dipenuhi (bernilai benar), dan tidak member pilihan aksi lain bila kondisi bernilai salah. Bentuk pemilihan yang lebih umum ialah memilih satu dari dua buah aksi bergantung pada nilai kondisinya

**If** kondisi **then**

**Aksi 1**

**Else**

**Aksi 2**

**Else** artinya “kalau tidak”. Bila kondisi bernilai benar aksi 1 akan dikerjakan, tetapi kalau tidak, aksi 2 yang akan dikerjakan. Misalnya pada pernyataan berikut :

**If** hari hujan **then**

Pergilah dengan naik becak

**Else**

Pergilah dengan naik motor

Jika kondisi “Hari hujan” bernilai benar, maka aksi “pergilah dengan naik beca” dilakukan, sebaliknya, aksi “pergilah dengan naik motor” akan dilakukan bila “hari hujan” tidak benar.

Contoh lainnya adalah menentukan nilai terbesar dari dua buah bilangan,  $x$  dan  $y$  (misalkan  $x \neq y$ ). andaikan bahwa pemroses dapat melakukan operasi membandingkan dua buah bilangan.

If  $x > y$  then

Tulis  $x$  sebagai bilangan terbesar

Else

Tulis  $y$  sebagai bilangan terbesar

Apabila pilihan aksi yang dilakukan lebih dari dua buah, maka struktur pemilihan menjadi lebih rumit, seperti pada contoh berikut (pemilihan bersarang) :

If lampu pengatur lalu lintas berwarna merah then

Anda harus berhenti

Else

If lampu lalu lintas berwarna kuning then

Anda boleh jalan tapi dengan hati-hati

Else

Anda boleh silakan terus berjalan

Contoh lain dari pemilihan bersarang adalah menentukan bilangan terbesar dari dua buah bilangan :  $x$ ,  $y$  , dan  $z$  :

If  $x > y$  then

If  $x > z$  then

Tulis  $x$  sebagai bilangan terbesar

Else

Tulis  $z$  sebagai bilangan terbesar

Else

If  $y > z$  then

Tulis  $y$  sebagai bilangan terbesar

Else

Tulis  $z$  sebagai bilangan terbesar

### 3. Pengulangan



Salah satu kelebihan computer adalah kemampuannya untuk mengerjakan pekerjaan yang sama berulang kali tanpa mengenal lelah, ini berbeda dengan manusia yang cepat lelah bila mengerjakan yang sama berulang-ulang, tidak hanya lelah, tetapi juga cepat bosan.

Sewaktu duduk di sekolah dasar, mungkin kalian pernah dihukum oleh guru untuk menuliskan sebuah kalimat sebanyak 500 kali, karena anda nakal atau tidak membuat PR. Misalkan kalimat tersebut adalah :

“Saya berjanji tidak akan nakal dan malas lagi”

Bila pekerjaan menulis kalimat ini diserahkan ke computer, maka pemrogram mungkin menuliskannya sebagai berikut :

Algoritma menulis 500 kalimat

Menulis kalimat “saya berjanji tidak akan nakal dan malas lagi”

Sebanyak 500 kali.

Deskripsi

1. Tulis “Saya berjanji tidak akan nakal dan malas lagi”

2. Tulis “saya berjanji tidak akan nakal dan malas lagi”

3. Tulis “saya berjanji tidak akan nakal dan malas lagi”

.

.

499. Tulis “saya berjanji tidak akan nakal dan malas lagi”

500. Tulis “saya berjanji tidak akan nakal dan malas lagi”

Tentu saja penulisan seperti ini merupakan pekerjaan yang dilarang oleh guru, karena instruksi

Tulis “saya berjanji tidak akan nakal dan malas lagi”

Harus kita tulis dalam teks algoritma sebanyak yang diperintahkan oleh Guru. Untuk mengatasi hal ini, maka penyusun algoritma menggunakan struktur pengulangan repeat-until (contoh perintah pengulangan, repeat artinya “ulangi” sedangkan until artinya “sampai”)

Sehingga algoritma menulis 500 buah kalimat dapat ditulis sebagai berikut :

Algoritma menulis 500 kalimat

Menulis kalimat “saya berjanji tidak akan nakal dan malas lagi”

Sebanyak 500 kali.

Deskripsi

Pencatat\_jumlah\_kalimat=0

Repeat

Tulis “Saya berjanji tidak akan nakal dan malas lagi”

Naikkan pencatat\_jumlah\_Kalimat dengan 1

Until pencatat\_jumlah kalimat = 500

Pemroses akan melaksanakan aksi tulis tersebut sebanyak 500 kali.

Struktur pengulangan ini dapat ditulis secara umum sebagai :

Repeat

Aksi

Until kondisi

Struktur pengulangan biasanya disebut (loop), dan bagian algoritma yang diulang (aksi) dinamakan (badan pengulangan/ loop body). Pengulangan dengan struktur repeat-until didasarkan pada kondisi berhenti, yaitu aksi di dalam badan loop dilaksanakan berulang-ulang sampai suatu kondisi tertentu dipenuhi (pada contoh di atas bila pencatat\_jumlah\_kalimat=500) bila kondisi tersebut tercapai, pengulangan aksi dihentikan.

Struktur repeat-until hanyalah satu dari beberapa struktur pengulangan. Masih ada beberapa struktur pengulangan lain yang tidak akan dibahas dalam bab ini. Seperti while-do, for, dan sebagainya. Yang akan dibahas pada bab-bab selanjutnya

## **Latihan Algoritma**

### **A. PILIHAN GANDA**

#### **Penjelasan**

#### **← (di isi dengan)**

1. Diberikan potongan algoritma sbb :

$i \leftarrow 1$   
output( $i + 2$ )  
Tampilan dari output( $i+2$ ) adalah  
a.  $i$                       b.  $i + 2$                       c.  $1 + 2$   
d. 3                      e. salah semua

2. Diberikan potongan algoritma sbb :

$i \leftarrow 1$   
 $i \leftarrow 2$   
output( $i$ )  
Tampilan dari output( $i$ ) adalah  
a. 1                      b. 2                      c. 1 dan 2  
d. 3                      e. salah semua

3. Diberikan potongan algoritma sbb :

$i \leftarrow 1$   
 $i \leftarrow i+2$   
output( $i$ )  
  
Tampilan dari output( $i$ ) adalah  
a. 1                      b. 2                      c.  $i$   
d. 3                      e. salah semua

4. Diberikan potongan algoritma sbb :

```
i ← 1  
j ← 2  
i ← i+j  
j ← i+j  
output(j)
```

Tampilan dari output(j) adalah

- a. j                      b. 3                      c. 5  
d. i+j                    e. salah semua

5. Diberikan potongan algoritma sbb :

```
i ← 1  
j ← 2  
i ← i - j  
j ← j - i  
output(j)
```

Tampilan dari output(j) adalah

- a. j                      b. 1                      c. 2  
d. 3                      e. salah semua

6. Diberikan potongan algoritma sbb :

```
A ← 5  
B ← 8  
B ← A  
A ← B  
Output(A,B)
```

Tampilan dari output(A,B) adalah

- a. 5 8                    b. 8 8                    c. 5 5  
d. 8 5                    e. salah semua

7. Diberikan potongan algoritma sbb :

```
A ← 5  
B ← 8  
C ← B  
B ← A  
A ← B  
Output(A,B)
```

Tampilan dari output(A,B) adalah

- a. 5 8                    b. 8 8                    c. 5 5  
d. 8 5                    e. salah semua

8. Diberikan potongan algoritma sbb :

```
Input(N)  
If N<0 then  
    N ← -N  
End if  
Output(N)
```

Jika dari perintah Input(N), dimasukkan data 5, maka Tampilan dari Output(N) adalah

- a. 0                      b. -5                      c. -10
- d. 5                      e. salah semua

9. Jika dari perintah Input(N) soal di atas, dimasukkan data -5, maka Tampilan dari Output(N) adalah

- a. 0                      b. -5                      c. -10
- d. 5                      e. salah semua

10. Diberikan potongan algoritma sbb :

```
Input(N)  
If N < 0 then  
    X ← -1  
Elseif N > 0 then  
    X ← 1  
Else  
    X ← 0  
End if  
Output(X)
```

Jika dari perintah Input(N), dimasukkan data 5, maka Output(X) menampilkan

- a. 0                      b. -5                      c. -1
- d. 5                      e. salah semua

## **B. ESSAY**

Buatlah sebuah urutan langkah (Algoritma sederhana dalam bahasa sehari-hari) untuk soal berikut sesuai dengan yang anda pikirkan :

- a. Perjalanan dari Bandung Ke Jakarta.
- b. Cara membuat Bala-bala.
- c. Supaya tidak dimarahi guru ketika tidak mengerjakan PR.
- d. 1. Menyatakan Cinta supaya diterima (Bagi Laki-laki)  
2. Menolak cinta supaya tidak tersinggung (Bagi Perempuan)

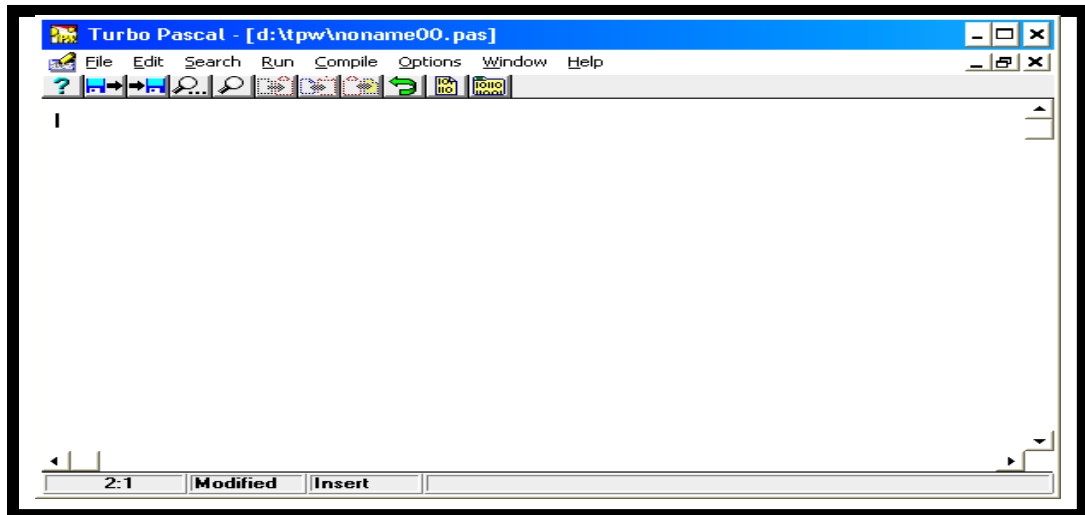
## BAB II ELEMEN DASAR TURBO PASCAL 1

### 1. Mengenal Editor Turbo Pascal

#### a. Memulai Turbo Pascal

Dari Menu Start ⇒ All Program ⇒ Pilih TPW 1.5 ⇒ TPW 1.5

Tampilan Menu Editor Turbo Pascal For Windows



Menu Utama digunakan untuk mengerjakan suatu perintah tertentu misalnya memanggil file, menyimpan file, mengkompilasi program.

#### b. Tombol Keyboard berkenaan dengan operasi kursor

- |                |   |   |
|----------------|---|---|
| < ↑ >          | : | Ke atas satu baris                      |
| < ↓ >          | : | Ke bawah satu baris                     |
| < ← >          | : | Ke Kiri satu karakter                   |
| < → >          | : | ke kanan satu karakter                  |
| < Home >       | : | Ke Kiri awal baris                      |
| < End >        | : | Ke kanan akhir baris                    |
| < Delete >     | : | Menghapus satu karakter ke kanan kursor |
| < Back Space > | : | Menghapus satu karakter ke kiri kursor  |
| < Page Up >    | : | Menggulung layar satu halaman atas      |
| < Page Down >  | : | Menggulung layar satu halaman ke bawah  |

#### c. Mengkompilasi Program

- 1) Pilih menu Compile
  - 2) Pilih submenu Compile (Alt + F9)
- Compile : Kegiatan memeriksa sintak kode program

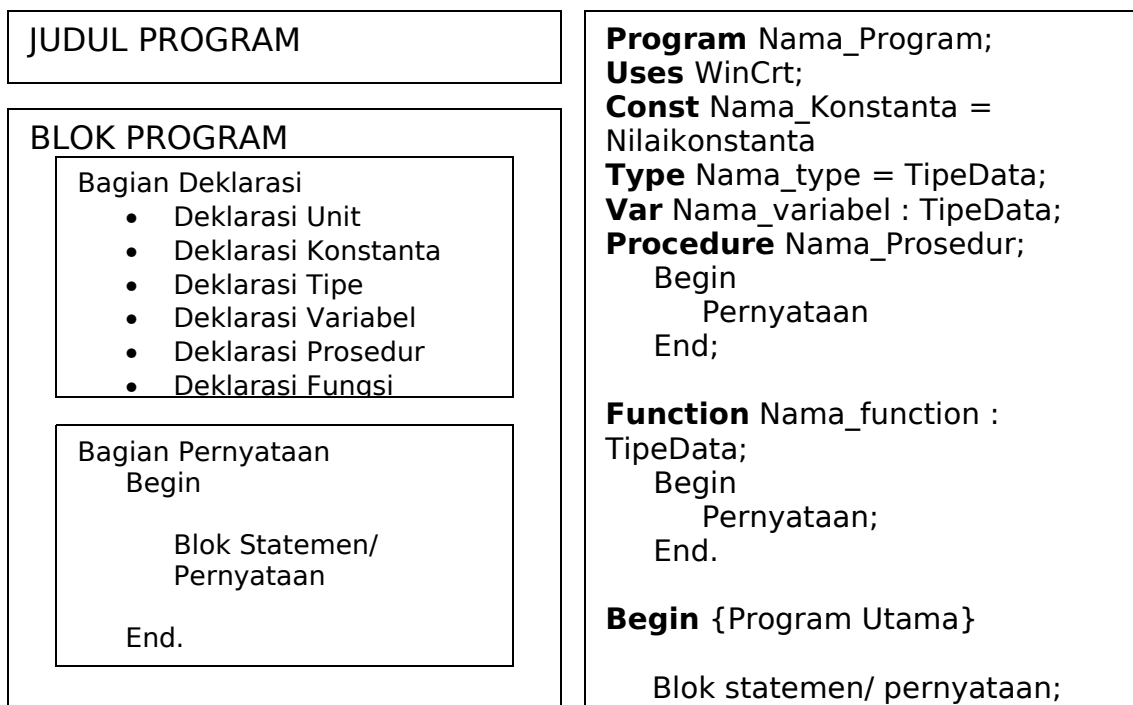
#### d. Menjalankan Program

- 1) Pilih menu Run
- 2) Pilih submenu Run (Ctrl + F9)

3) Hasil program ditampilkan, untuk kembail ke desain program tekan tombol Enter

- e. Menyimpan Program
  - 1) Pilih menu File
  - 2) Pilih Submenu Save (F2)
  - 3) Ekstensi untuk file program yang dibuat dalam Turbo Pascal adalah .PAS
- f. Menutup Program
  - 1) Pilih menu window
  - 2) Pilih submenu close (Alt + F3)
- g. Mengakhiri Program
  - 1) Pilih menu File
  - 2) Pilih submenu Exit (Alt + X)
- h. Membuka File
  - 1) Pilih menu File
  - 2) Pilih submenu Open ...(F3)

## 2. Struktur Program Pascal



### a. Judul Program

Program pada bagian deklarasi merupakan kata yang berguna untuk memberikan judul program yang akan dibuat.

Syarat untuk nama program

- Antara kata program dan nama program boleh dipisahkan dengan spasi, enter dan lain-lain.
- Nama program harus merupakan suatu kesatuan dan tidak boleh dipisah (spasi)
- Tidak boleh terdapat tanda-tanda aritmetika seperti "+", "-", "\*", "/", dan lain-lain.
- Boleh menggunakan garis bawah.



- Tidak boleh dimulai dengan suatu bilangan.
- Diakhiri tanda;  
 Contoh            1) coba\_coba; (Benar)  
                       2) 2unlimited; (Salah)

## **b. Uses**

Uses merupakan perintah yang digunakan untuk memakai suatu unit tertentu dari turbo pascal.

Unit → Kumpulan instruksi yang merupakan procedure dan fungsi yang dipakai dalam turbo pascal.

Pada turbo pascal semua unit harus menggunakan kata uses, kecuali unit system.

### **1) Unit system**

Unit system merupakan unit yang langsung terpasang pada Turbo Pascal sehingga untuk menggunakan perintah-perintah yang terdapat di dalam unit system ini tidak perlu menggunakan perintah uses.

#### **Perintah Unit System :**

Break	Abs	ArcTan
Continue	Cos	Exp
...		...
Close	EOF	EOLN

### **2) Unit WinCrt**

Unit Crt merupakan unit yang mengatur kerja layar dan keyboard atau I/O. Sebagai contoh, perintah CLRSCR yg digunakan untuk membersihkan layar tidak akan berjalan tanpa menggunakan perintah Uses WinCrt; jadi syarat mutlak untuk menggunakan perintah-perintah yang terdapat di dalam unit Crt adalah menggunakan kata "Uses WinCrt"

#### **Perintah Unit Crt:**

AssignCrt	ClrEol	ClrScr
Delay	DelLine	GotoXY
...		...
WhereX	WhereY	Readkey

### **3) Unit Dos**

Unit Dos merupakan unit yang bekerja di dalam system Dos

#### **Perintah Unit Dos:**

GetDate	SetDate	GetTime
PackTime	UnPackTime	setFtime
...		...
setCbreak	SetVerify	GetVerify

### **4) Unit WinDos**

Unit WinDos merupakan unit yang bekerja di dalam system Windows

#### **Perintah Unit WinDows:**

GetDate	GetFTime	GetTime
PackTime	SetDate	SetFtime
...		...
GetCurDir	RemoveDir	SetCurDir

## 5) Unit String

Unit String merupakan unit yang bekerja untuk teks-teks yang berada pada Pascal

### Perintah Unit String:

StrCat	StrComp	StrCopy
StrECopy	StrlComp	StrlLCat
...		... ..
StrRScan	StrScan	StrUpper

## 6) Unit Graph

Unit Graph adalah unit yang berorientasi ke pemrograman grafik.

**Perintah Unit Graph:**

Arc	Bar	Bar3D
Circle	ClearDevice	ClearViewPort
...		...
TexHeigh	TextWidth	SetWriteMode

## 7) Unit Overlay

Unit overlay adalah unit yang mengatur pembagian sebuah program menjadi 2 file atau lebih.

**Perintah Unit Overlay:**

OvrClearBuf	OvrGetBuf	OvrGetRetry
Ovrlunit	OvrlInitEMS	OvrSetBuf
OvrsetRetry		

## 8) Unit Printer

Unit Printer adalah unit yang mengatur kerja printer sehingga dapat dilakukan pengiriman langsung ke printer. Kata yang merupakan pengganti operasi ke printer adalah Ist. Jadi bila diberikan kata Ist tanpa menggunakan Uses Printer; maka proses ke printer tidak akan berjalan.

### c. Type

Type adalah perintah yang berguna untuk mendefinisikan suatu type yang akan dimasukkan ke dalam suatu variable.

### d. Const

Perintah Const adalah perintah untuk mendefinisikan suatu konstanta

Tata cara penulisan konstanta adalah sebagai berikut :

- Menuliskan Const pada awal pendefinisian konstanta
- Konstanta tidak hanya berlaku untuk bilangan tetapi bisa juga berupa suatu string.
- Dalam konstanta boleh terdapat operasi aritmatika perkalian, pembagian, penambahan dll.
- Nama konstanta tidak mengandung spasi dan tanda aritmetika.

### e. Var

Kata var merupakan tanda dimulainya pendefinisian variable-variable yang akan digunakan pada program.

Tata cara penulisan

- Pendefinisian variable dimulai dengan kata Var dan memperhatikan tata cara penulisan pendefinisian variable, yaitu :  
Nama\_var1, nama\_var2, ... tipe variable
- Nama variable tidak boleh mengandung spasi dan tanda aritmatik, serta tidak boleh dimulai dengan bilangan.
- Nama variable tidak boleh sama dengan nama perintah

## Tipe Variabel

### 1) Numerik

#### a) Bilangan Bulat

Tipe	Range
Shortint	-128 - +127
Integer	-32.768 - +32.767
Longint	-2147483648 - +2147483647
Byte	0 - 255
Word	0 - 65535

Perbedaan masing-masing tipe terletak pada jangkauan nilainya.

#### b) Bilangan Real

Tipe	Range	Ketelitian (Digit)	Bytes
Real	$2,9 \times 10^{-39} - 1,7 \times 10^{38}$	11 - 12	6
Single	$1,5 \times 10^{-45} - 3,4 \times 10^{38}$	7 - 8	4
Double	$5,0 \times 10^{-324} - 1,7 \times 10^{308}$	15 - 16	8
Extended	$3,4 \times 10^{-4932} - 1,1 \times 10^{4932}$	19 - 20	10
Comp	$-2^{63} + 1 - 2^{63} - 1$	19 - 20	8

### 2) Non Numerik

#### a) Tipe String

Tipe string adalah tipe yang hanya terdiri dari kata-kata.

#### b) Tipe Char

Tipe yang variabelnya hanya dapat menampung satu karakter saja.

#### c) Tipe Boolean

Tipe variable yang hanya menampung dua kemungkinan, yaitu False atau True.

## f. Operator

Sebelum menyusun program yang mengolah data dan variable, anda harus mengenal operator-operator di dalam Pascal. Operator adalah symbol-simbol atau kata-kata yang dipakai untuk melaksanakan suatu operasi terhadap data dan variable. Secara table, maka operator-operator tersebut dapat dikelompokkan sebagai berikut :

Jenis Operator	Simbol	Keterangan
Operator Negasi	-	Mengubah data angka menjadi harga negatifnya
Operator Matematik	+	Penjumlahan
	-	Pengurangan
	*	Perkalian

	<b>/</b>	Pembagian
	<b>:=</b>	Sama dengan
	<b>Div</b>	Pembagian integer (sisanya dibuang)
	<b>Mod</b>	Sisa pembagian

<b>Jenis Operator</b>	<b>Simbol</b>	<b>Keterangan</b>
Operator Relasional	<	Kurang dari
	>	Lebih dari
	=	Sama dengan
	<=	Kurang dari atau sama dengan
	>=	Lebih dari atau sama dengan
	<>	Tidak sama dengan
Operator Boolean	<b>Not</b>	Operasi-operasi logika yang dilakukan terhadap nilai-nilai Boolean
	<b>And</b>	
	<b>Or</b>	
	<b>Xor</b>	
Operator Bit	<b>Shl</b>	Geser satu bit ke kiri
	<b>Shr</b>	Geser satu bit ke kanan
	<b>Not</b>	Komplemen suatu bit
	<b>And</b>	Operasi and terhadap dua bit
	<b>Or</b>	Operasi or terhadap dua bit
	<b>Xor</b>	Operasi xor terhadap dua bit
Operator Keanggotaan himpunan	<b>In</b>	Menyatakan keanggotaan himpunan
Operator address	@	Menunjukkan addres memory suatu variable
		Menyatakan nilai yang ditunjukkan oleh pointer

#### **g. Label**

Label adalah pendeklarasian kata-kata atau tanda yang digunakan untuk menyatakan suatu tempat pada program. Label ini dapat berisi apa saja, kecuali tanda-tanda operasi aritmetik, seperti +, - dan lain-lain.

## LATIHAN

### Pilihan Ganda

1. Untuk mengkompilasi program digunakan perintah :
  - a. F9
  - b. Ctrl + F9
  - c. Alt + F9**
  - d. Ctrl + F8
  - e. Alt + F8
  
2. Untuk menjalankan program digunakan
  - a. F9
  - b. Ctrl + F9**
  - c. Alt + F9
  - d. Ctrl + F8
  - e. Alt + F8
  
3. Perintah F3 (Open) berada pada menu
  - a. File**
  - b. Edit
  - c. Search
  - d. Run
  - e. Compile
  
4. Menutup program digunakan perintah
  - a. Ctrl + F3
  - b. F3
  - c. Alt + F3**
  - d. Ctrl + F2
  - e. Alt + F2
  
5. Manakah identifier yang benar untuk menamakan nama program :
  - a. prog-a
  - b. prog\_a**
  - c. prog+a
  - d. prog//a
  - e. Salah semua
  
6. Manakah ekspresi yang tidak dapat dilakukan oleh tipe data real
  - a.  $a := a + b$
  - b.  $a := a - b$
  - c.  $a := a * b$
  - d.  $a := a \bmod b$**

e. Salah semua

7. Tipe data di bawah ini mana yang tidak dapat melakukan operasi aritmatika

- a. integer
- b. byte
- c. real
- d. Boolean
- e. Salah semua

8. Yang memiliki hierarchi paling tinggi di antara operasi logika AND, OR dan NOT adalah

- a. AND
- b. NOT
- c. OR
- d. semua sama
- e. Salah semua

9. Yang paling rendah dari operasi logika AND, OR dan NOT adalah :

- a. AND
- b. NOT
- c. OR
- d. semua sama
- e. salah semua

10. Manakah ekspresi yang tidak dapat dilakukan oleh tipe data integer

- a.  $x := x + 5$
- b.  $y := y - z$
- c.  $k := k / 5$**
- d.  $r := r * s$
- e. Salah semua

11. Unit yang langsung terpasang pada Turbo Pascal sehingga untuk menggunakan perintah-perintah yang terdapat di dalam unit system ini tidak perlu menggunakan perintah uses adalah

- a. Unit WinCrt
- b. Unit System**
- c. Unit WinDos
- d. Unit String
- e. Unit Dos

12. Unit String adalah

- a. Unit yang langsung terpasang pada Turbo Pascal sehingga untuk menggunakan perintah-perintah yang terdapat di dalam unit system ini tidak perlu menggunakan perintah uses
- b. Unit yang bekerja di dalam system Windows
- c. Unit yang bekerja di dalam system Dos



- d. Unit yang mengatur kerja layar dan keyboard atau I/O.
- e. Unit yang bekerja untuk teks-teks yang berada pada Pascal.**

13. Perintah yang berguna untuk mendefinisikan suatu type yang akan dimasukkan ke dalam suatu variable adalah....
- Type
  - Const
  - Var
  - Operator
  - Label
14. Range -32.768 sampai + 32.767 adalah tipe variable .....
- Shortint
  - Integer
  - Longint
  - Byte
  - Word
15. Word adalah tipe variable dengan range ....
- 128 sampai +127
  - 32.768 sampai + 32.767
  - 0 - 255
  - 0 - 65535
  - salah semua
16. Nilai dari 23 div 5 adalah
- 2
  - 3
  - 4
  - 5
  - 6
17. Nilai dari 23 mod 5 adalah
- 2
  - 3
  - 4
  - 5
  - 6
18. Manakah yang salah dari deklarasi di bawah ini
- program a; const k=10;**
  - program abc ; const p:6;
  - program pqr ; const r:=8;
  - program satu; const x>y;

19. Manakah yang benar dari deklarasi di bawah ini

a. **program abc;**

**var x, y : real;**

b. program pqr;

var x,y=byte;

c. program xyz ;

var x:=boolean;

d. program klm;

var z,y=char;

e. Salah semua

20. Manakah yang salah dari deklarasi di bawa ini :

**a. var a = array[1..5] of integer ;**

b. var a : array[0..5] of byte ;

c. var a : array[2..6] of byte ;

d. var a : array[1..5] of real ;

e. Salah semua

## ESSAY

1. Apa saja yang terdapat pada bagian deklarasi suatu Program Turbo Pascal

2. Apakah yang dimaksud dengan kata-kata di bawah ini, pada bagian deklarasi program pascal.

a. Program

b. Const

c. Integer

d. Double

e. Label

3. Tuliskan nama-nama/identifier variable untuk menyimpan data-data:

a. Nama murid kelas 3A

b. Jumlah guru di sekolah

c. Banyaknya tabungan saat ini.

d. Luas silinder

e. Banyaknya kaset yang disewa

4. Misalkan anda mendeklarasikan variable sebagai berikut :

Var

a, b, c : Integer;

x, y, z : Real;

d, e : char;

p, q : Boolean;

i : Longint;

```
r      : String[5];  
s      : String[32];
```

Pernyataan manakah yang tidak dapat dipakai di dalam program utama anda !

- |             |                     |
|-------------|---------------------|
| a.          | f.                  |
| c := a + b; | x := y * z;         |
| b.          | g.                  |
| c := a/b;   | d := 'b';           |
| c.          | h.                  |
| c := a - b; | e := #31;           |
| d.          | i.                  |
| x := y + z; | p := True;          |
| e.          | j.                  |
| x := y/z;   | q := True or False; |

5. Apabila dideklarasikan :

Var

A, B, C : Real;

X, Y, Z : Integer;

P : String [40];

H : Char;

Dan diketahui

A = 3.2 B = 6.0 C = 1.5

X = 4 Y = 63 Z = 17

Apa hasil dari ekspresi-ekspresi berikut ini :

- |    |                |    |                  |
|----|----------------|----|------------------|
| a. | A := 0.5 * A   | f. | P := 'Selamat' + |
| b. | B := A/4       |    | 'Belajar'        |
| c. | X := X + 1     | g. | H := #65         |
| d. | Y := X * 2     | h. | H := 'Z'         |
| e. | C := C - A - B | i. | X := 10 div 4    |
|    |                | j. | Z := 11 mod 4    |

6. Ubahlah rumus-rumus berikut ini menjadi ekspresi yang sesuai dengan Pascal :

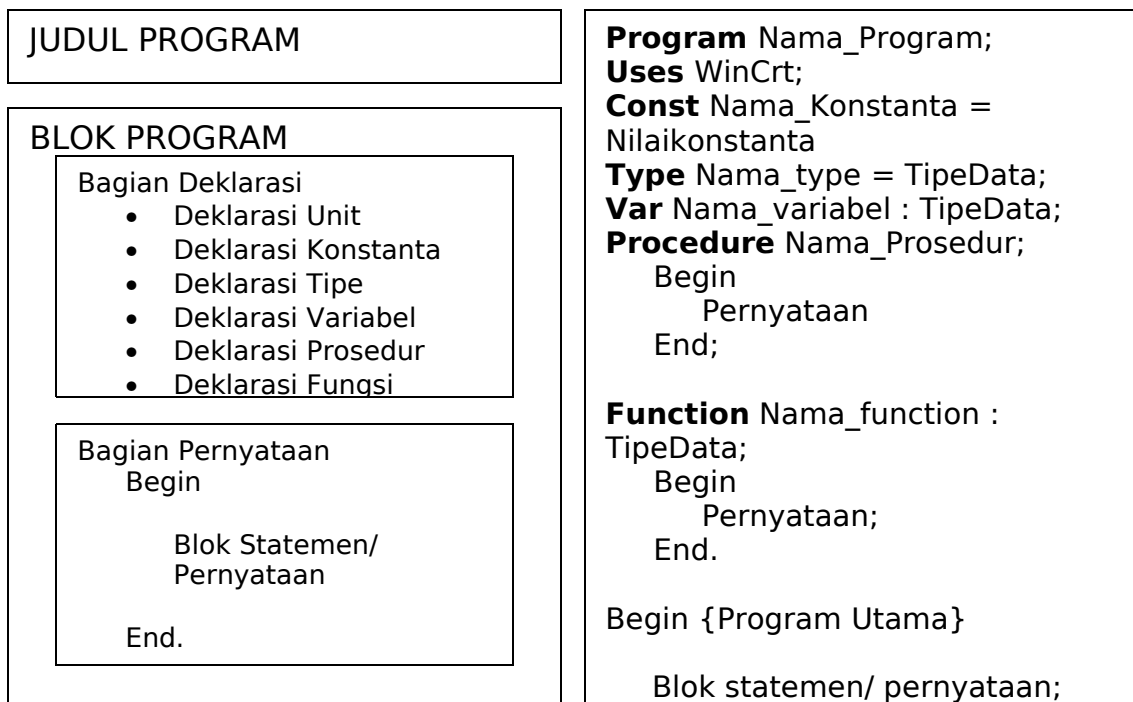
- $y = mx + c$
- $x = vt + 0.5at^2$
- $w = \frac{x+y}{45}$
- $n = \frac{x}{p-3q} - \frac{a+b}{p+2q}$
- $z = 13t^4 + 10t^3 + 5t^2 + 10$
- $j = (k + 3)j$

7. Buatlah program untuk menghitung luas sebuah segitiga yang memiliki alas 12.50 dan tinggi 50. Pergunakan deklarasi Var.

8. Buatlah program yang dipergunakan untuk menghitung honorarium seorang manajer pada akhir bulan. Dalam satu bulan, ia bekerja 25 hari. Setiap hari ia bekerja 6 jam. Setiap jam ia mendapatkan Rp. 15.000,- dan ia kena pajak pendapatan sebesar 10%. Pergunakan deklarasi VAR dan CONST apabila perlu. Jalankan, lalu simpan ke Flash Disk.

## BAB III ELEMEN DASAR TURBO PASCAL 2

### 3.1. Struktur Program Pascal



### 3.2. INPUT/ OUTPUT

#### 3.2.1. Write dan Writeln

Perintah Write dan Writeln adalah perintah yang berguna untuk mencetak kata-kata atau variable ke layar komputer. Contoh untuk

mencetak kata 'Sedang belajar pascal', perintah yang dapat diberikan adalah :

Write ('sedang belajar pascal') Atau Writeln ('sedang belajar pascal')

### **Praktek 1 (Ketikkan dalam program Pascal)**

**Program** Mencetak1;

**Uses** WinCrt;

**Begin**

Write ('sedang belajar pascal');

**End.**

## Praktek 2 (Ketikkan dalam program Pascal)

```
Program Mencetak2;  
Uses WinCrt;
```

```
Begin
```

```
    Write ('sedang belajar pascal');  
    Writeln ('sedang belajar pascal');  
    Write ('sedang belajar pascal');  
    Writeln ('sedang belajar pascal');  
    Writeln ('sedang belajar pascal');  
    Writeln ('sedang belajar pascal');
```

```
End.
```

## Praktek 3 (Ketikkan dalam program Pascal)

```
Program Mencetak3;  
Uses WinCrt;  
Const  
    A : Integer = 100;
```

```
Begin
```

```
    Write (A);
```

```
End.
```

## Praktek 4 (Ketikkan dalam program Pascal)

```
Program Mencetak4;  
Uses WinCrt;  
Const  
    A : Integer = 100;
```

```
Begin
```

```
    Write ('Nilai yang terdapat di A adalah',A);
```

```
End.
```

Hasil yang tampak pada layar adalah “Nilai yang terdapat di A adalah : 100)

Pada layar hasil akan tercetak angka 100, yang merupakan hasil dari proses program di atas.

Dari contoh program di atas dapat disimpulkan beberapa hal mengenai

**Write** dan **Writeln**, yaitu :

- Segala sesuatu yang hendak di cetak harus terdapat diantara kurung buka [ ( ] dan kurung tutup [ ) ].
- Bila ingin mencetak kata-kata, kata tersebut harus terdapat di antara tanda petik satu ('), sedang untuk mencetak variable tidak perlu diberikan tanda petik satu (').



- Untuk setiap pemakaian `Write` dan `WriteLn` harus diakhiri dengan tanda titik koma (;).

### 3.2.2. Komentar

Komentar adalah bagian program yang tidak akan diproses oleh TurboPascal. Komentar ini hanya digunakan sebagai dokumentasi saja. Untuk memulai suatu komentar, harus memberikan tanda { dan diakhiri }

### 3.3. Read dan Readln

Memang tidak ada perbedaan yang jelas antara perintah read dan readln. Perbedaan read dan readln ini akan terdapat pada operasi file, sedang perbedaan bila input dari standar input atau keyboard adalah pada pemasukan Enter. Read tidak akan menerima Enter, sedangkan Readln akan menerima Enter sebagai masukan.

#### Praktek 5 (Ketikkan dalam program Pascal)

```
Program ambil1;  
Uses WinCrt;  
Var  
    A : Integer;  
  
Begin  
    Write ('Masukan suatu angka :' );  
    Read (A);  
    Writeln ('Angka yang baru Anda masukan adalah : ', A)  
End.
```

#### Praktek 6 (Ketikkan dalam program Pascal)

```
Program ambil2;  
Uses WinCrt;  
Var  
    Nama : string;  
  
Begin  
    Write ('Masukan Nama Anda :' );  
    Read (A);  
    Writeln ('Selamat Belajar Pascal ', A)  
End.
```

#### Praktek 7 (Ketikkan dalam program Pascal)

```
Program ambil3;  
Uses WinCrt;  
Var  
    Nilai : Real;  
    Nama : String;
```

**Begin**

```
Write ('Masukan Nama dan Nilai Anda dengan diselingi spasi' );  
Readln (Nama, nilai);  
Writeln ('Saudara ',Nama,'Anda mendapat nilai ',nilai);
```

**End.****3.4. Readkey**

Readkey adalah fungsi masukan yang hanya akan mengambil satu karakter masukan. Jenis variable yang dapat diinput oleh Readkey pun hanya jenis karakter.

**Contoh Praktek 7 (Ketikkan dalam program Pascal)**

```
Program Reky1;
```

```
Uses WinCrt;
```

```
Var
```

```
    Ch : Char;
```

**Begin**

```
    Writeln ('Tekan Karakter apa saja:');
```

```
    Ch := Readkey;
```

```
    Writeln ('Karakter yang baru Anda tekan adalah ',Ch)
```

**End.**

## LATIHAN

### PILIHAN GANDA

1. Perintah yang digunakan untuk mencetak kata-kata atau variable ke layar computer adalah...

- a. Readln
- b. Writeln
- c. Input
- d. Output
- e. Salah semua

2. Perhatikan Program di bawah

**Program** soal2;

**Uses** WinCrt;

**Begin**

Writeln ('sedang belajar pascal');

Write ('sedang belajar pascal');

Writeln ('sedang belajar pascal');

**End.**

Hasilnya adalah....

- a. sedang belajar  
pascal sedang  
belajar pascal  
sedang belajar  
pascal
- b. sedang belajar  
pascal

sedang belajar pascal  
sedang belajar pascal

- c. sedang belajar  
pascal

sedang belajar pascal sedang belajar pascal

- d. sedang belajar  
pascal sedang  
belajar pascal

sedang belajar pascal

- e. Salah semua

3. Perhatikan Program di bawah

**Program** soal3;

**Uses** WinCrt;

**Begin**

Write ('sedang belajar pascal');

Writeln ('sedang belajar pascal');

Writeln ('sedang belajar pascal');

**End.**

Hasilnya adalah....

- |  |  |  |
|--|--|--|
|  | a. sedang belajar pascal sedang belajar pascal sedang belajar pascal |  |
|  | b. sedang belajar pascal   |  |
| sedang belajar pascal<br>sedang belajar pascal |  | c. sedang belajar pascal                       |
|  |  | d. sedang belajar pascal sedang belajar pascal |
| sedang belajar pascal sedang belajar pascal    |  | e. Salah semua                                 |

4. Perhatikan program di bawah

1) **Program soal4**

2) **Uses** WinCrt;

3) **Const**

A : Integer = 100;

4) **Begin**

Write ('A');

5) **End.**

Pada soal nomor 4 terdapat kesalahan yaitu pada nomor

- |    |    |
|----|----|
| a. | d. |
| 1  | 4  |
| b. | e. |
| 2  | 5  |
| c. |    |
| 3  |    |

5. Apabila soal nomor 4 telah diperbaiki maka hasilnya adalah :

- |     |             |
|-----|-------------|
| a.  | d.          |
| 100 | A           |
| b.  | e.          |
| 200 | Salah semua |
| c.  |             |
| 300 |             |

6. Lambang komentar adalah

- |    |     |
|----|-----|
| a. | c.  |
| ;  | { } |
| b. | d.  |
| :  | ( ) |

e.  
" "

7. Readkey adalah

- a. Fungsi masukan yang hanya akan mengambil satu karakter masukan.
- b. Fungsi masukan yang boleh mengambil beberapa karakter
- c. Fungsi Keluaran yang mengambil satu karakter masukan
- d. Fungsi Keluaran yang boleh mengambil beberapa karakter
- e. Salah semua

8. Perhatikan Program di bawah

```
Program soal8;  
Uses WinCrt;  
Begin  
Write('pascal teh lieur');  
      Write('pascal teh lieur');  
Write('pascal teh lieur');  
End.  
Hasilnya adalah....
```

pascal teh lieur  
pascal teh lieur

pascal teh lieur

pascal teh lieur  
pascal teh lieur

- a. pascal teh lieur  
pascal teh lieur  
pascal teh lieur
- b. pascal teh lieur
- c. pascal teh lieur  
pascal teh lieur
- d. pascal teh lieur
- e. Salah semua

9. Perhatikan Program di bawah

```
Program soal9;  
Uses WinCrt;
```

**Begin**

```
Write('pascal teh lieur');
      Write('pascal teh lieur');
Write('pascal teh lieur');
```

**End.**

Hasilnya adalah....

```
pascal teh lieur
pascal teh lieur
```

```
pascal teh lieur
```

```
      pascal teh lieur
pascal teh lieur
```

- a. pascal    teh    lieur  
pascal    teh    lieur  
pascal teh lieur
- b. pascal teh lieur

- c. pascal    teh    lieur  
pascal teh lieur

- d. pascal teh lieur

- e. Salah semua

# 10. Perhatikan Program di bawah

**Program** soal9;

**Uses** WinCrt;

**Begin**

```
Writeln('pascal teh lieur');
Writeln('      pascal teh lieur');
Writeln('pascal teh lieur');
```

**End.**

Hasilnya adalah....

```
pascal teh lieur
pascal teh lieur
```

```
pascal teh lieur
```

```
      pascal teh lieur
pascal teh lieur
```

- a. pascal    teh    lieur  
pascal    teh    lieur  
pascal teh lieur
- b. pascal teh lieur

- c. pascal    teh    lieur  
pascal teh lieur

- d. pascal teh lieur

- e. Salah semua

## ESSAY

1. Buatlah program untuk mencetak tulisan seperti di bawah ini :

```
Turbo Pascal
  Turbo Pascal
    Turbo Pascal
      Turbo Pascal
        Turbo Pascal
```

2. Buatlah program yang dapat menampilkan seperti di bawah ini :

```
*****  *****
*****  *****
*****  *****
*****
*****
*****  *****
*****  *****
*****  *****
```

3. Buatlah program dengan contoh hasil eksekusi sebagai berikut :

```
Nama                : Inge          (di input) ↵
Nama Keluarga       : Sudjono      (di input) ↵
Nama Lengkap Anda adalah : Inge Sujono
```

4. Buatlah program sebagai berikut :

- Menghitung luas sebuah segitiga dengan input data berupa panjang alas, dan tinggi segitiga tersebut.
- Menghitung luas lingkaran dengan input data berupa keliling lingkaran tersebut.
- Menghitung laba jika diketahui harga penjualan dan harga pembelian
- Menghitung diskon jika diketahui harga penjualan sebelum discout dan harga penjualan setelah discout.

5. Tuliskan bentuk-bentuk ekspresi ini dalam bahasa Pascal :

f.  $x = 5$

g.  $z = a * b^2 - t^3$

h.  $w = \frac{x + 3y}{p - 4t}$

i.  $g = \frac{a - 10p}{5 - 2r} - \frac{p - r}{8a + p}$



6. Apakah hasil dari ekspresi-ekspresi Boolean berikut ini apabila diketahui  $x = 5$ ,  $y = \text{'Arlene'}$ ,  $I = 4$ ,  $J = 7$ ,  $K = 12$ ,  $L = 19$ .

- |    |  |
|----|--|
| a. | $10 > 5$   |
| b. | $x * 2 + 3 > x * 5$  |
| c. | $y <> \text{'Lala'}$   |
| d. | $(Y = \text{'Arlene'}) \text{ or } (Y = \text{'Lala'})$            |
| e. | $(\text{not}(X \geq 5)) \text{ or } (Y = \text{'Lala'})$           |
| f. | $(J < 10) \text{ or } (K = 7) \text{ and } (L > 10)$               |
| g. | $(J > 10) \text{ or } (K = 7) \text{ and } (K <> 10)$              |
| h. | $\text{not } (I < J) \text{ or } ((I <> K) \text{ and } (K <> L))$ |
| i. | $\text{not } (I < J) \text{ or } (I <> K) \text{ and } (K <> L)$   |
| j. | $(\text{non } (I < J) \text{ or } (I <> K)) \text{ and } (K <> L)$ |

7. Buatlah sebuah program untuk menghitung keliling suatu segitiga yang memiliki sisi berturut-turut  $a = 10$ ,  $b = 25$  dan  $c = 30$ .

8. Buatlah program yang akan menghitung panjang sisi miring suatu segitiga siku-siku. Panjang kedua sisi yang lain ditanyakan oleh program. Gunakan fungsi `sqrt` untuk menghitung akar suatu bilangan, dan fungsi `sqr` untuk menghitung kuadrat (pangkat dua) suatu bilangan.

## BAB IV PENYELEKSIAN KONDISI

### 4.1. Kondisi

Untuk menyatakan kondisi, diperlukan dua buah besaran yang dibandingkan. Untuk membandingkan kondisi ini diperlukan tanda-tanda pembanding.

**= (sama dengan) ; > (Lebih besar) ; < lebih kecil ; > = (lebih besar atau sama dengan) ; < = (lebih kecil atau sama dengan) ; < > tidak sama dengan.**

Bila suatu pernyataan kondisi memerlukan dua kondisi, dapat digabungkan dengan operator Or dan And.

Kondisi And (dan)		
Kondisi 1	Kondisi 2	Hasil
Benar	Benar	Benar
Benar	Salah	Salah
Salah	Benar	Salah
Salah	Salah	Salah

Kondisi Or (atau)		
Kondisi 1	Kondisi 2	Hasil
Benar	Benar	Benar
Benar	Salah	Benar
Salah	Benar	Benar
Salah	Salah	Salah

### 4.2. IF ... Then

Kondisi **if ... then** merupakan kondisi yang paling sederhana.

***If kondisi then proses untuk kondisi benar;***

**Contoh Praktek 1 (Ketikkan dalam program Pascal)**

```
Program If_01;  
uses WinCrt;  
Var  
  A, B : Integer;  
  
Begin  
  
  write ('Masukkan Angka Pertama : ');  
  readln(A);  
  write ('Masukkan Angka Kedua : ');  
  readln(B);  
  If A > B then  
    Write ('A lebih besar dari B');  
  End.
```

### 4.3. If ... then ... else

Bila if...then hanya menyatakan kondisi yang benar, pernyataan kondisi if...then...else dapat menampung dua kondisi.

**If kondisi then kondisi benar else kondisi salah;**

### **Contoh Praktek 2 (Ketikkan dalam program Pascal)**

```
Program If_02;  
uses WinCrt;  
Var  
  A, B : Integer;  
  
Begin  
  
  write ('Masukkan Angka Pertama : ');  
  readln(A);  
  write ('Masukkan Angka Kedua : ');  
  readln(B);  
  If A > B then  
    Write ('A lebih besar dari B');  
  Else  
    Write('B lebih besar dari A');  
  End.
```

#### **4.4. If ... then bertingkat**

Maksud If..then bertingkat adalah perintah if...then dengan if dalam suatu if. Tingkatan ini ditentukan oleh tanda titik koma (;) yang merupakan akhir dari suatu kalimat.

### **Contoh Praktek 3 (Ketikkan dalam program Pascal)**

```
Program If_03;  
uses WinCrt;  
Var  
  Suhu : Integer;  
  
Begin  
  write ('Masukkan Nilai Suhu: ');  
  readln(Suhu);  
  If Suhu < 0 then  
    Write ('Bentuk Benda Beku')  
  Else If (Suhu>=0) And (Suhu<=100) then  
    Write('Bentuk Benda Cair')  
  Else  
    Write('Bentuk Benda Gas');  
  End.
```

### **Contoh Soal**

1. Buatlah program yang menanyakan Nilai jika Input :  
antara 0 – 69 Muncul (Nilai Kurang perlu remedial)  
antara 70 – 100 Muncil (Nilai Sudah Bagus tidak remedial)

selain di atas muncul (Anda salah memasukkan nilai coba ulangi Nilai antara 0 – 100)

2. Buatlah program pada nilai kuliah. Dengan persyaratan berikut:

- Jika nilai lebih dari 60 anda lulus mata kuliah ini
- Jika nilai lebih dari sama dengan 85 anda mendapat nilai A
- Jika nilai lebih dari sama dengan 70 anda mendapat nilai B
- Jika nilai lebih dari sama dengan 60 anda mendapat nilai C
- Jika kurang dari 60 anda tidak lulus

3. Buatlah program yang menanyakan umur seseorang. Apabila umurnya kurang dari 17 tahun, maka dicetak tulisan 'Cuci kaki, lalu tidur saja yaa...'. Apabila umurnya 17 hingga 45 tahun, dicetak tulisan 'selamat menyaksikan...', apabila umurnya lebih dari 45 tahun, dicetak tulisan 'sebaiknya opa istirahat saja...'.

#### 4.5. Case ... Of

Case...Of adalah pemecahan dari if bertingkat. Pada Case...Of dapat dilakukan perbandingan beberapa kondisi dari suatu variable.

##### **Case variable of peluang; end;**

Dimana variable adalah variable yang akan diperiksa isinya; peluang adalah kemungkinan isi variable dan apa yang dilakukan jika peluang tersebut benar. Setiap peluang harus diakhiri dengan tanda titik koma (;). Jangan lupa memberikan kata End; pada akhir case.

#### **Contoh Praktek 4 (Ketikkan dalam program Pascal)**

1. program MenuRestoran;
- 2.
3. Uses
4. Crt;
- 5.
6. Var
7. pilih : integer;
- 8.
9. begin
10. writeln('Menu Restoran Junk Food');
11. writeln('-----');
12. writeln('1. Cireng gila');
13. writeln('2. Bala-bala gelo');
14. writeln('3. Gehu garelo');
15. write('Pilih pilihan anda : ');
16. readln(pilih);

```

17. case pilih of
18.   1 : writeln('Selamat Menikmati Cireng Gila');
19.   2 : writeln('Selamat Menikmati Bala-bala Gelo');
20.   3 : writeln('Selamat Menikmati Gegu garelo');
21. else
22.   writeln('Salah Input! Masukkan angka 1, 2, atau 3!!');
23. end;
24. readln;
25. end.

```

### Contoh Praktek 5 (Ketikkan dalam program Pascal)

**Program Case\_02;**

**Uses wincrt;**

**Var**

**Ch : char;**

**Begin**

Write ('masukkan suatu karakter : ');

Ch := Readkey;

Case Ch of

  'A' : Writeln ('Anda menekan A');

  'B' : Writeln ('Anda menekan B');

  'C' : Writeln ('Anda menekan C');

End;

**End.**

### Contoh Praktek 6 (Ketikkan dalam program Pascal)

**Program Case\_03;**

**Uses wincrt;**

**Var**

**I : Integer;**

**Begin**

ClrScr;

Write ('masukkan suatu bilangan ');

Readln (I);

Case I of

  1, 2, 3, 4, 5, 6, 7, 8, 9, 10 :

    Writeln ('Anda menekan antara 1 hingga 10');

  11,12,13,14, 15, 16, 17, 18, 19, 20 :

    Writeln ('Anda menekan antara 11 hingga 20');

  21, 22, 23, 24, 25, 26, 27, 28, 29, 30 :

    Writeln ('Anda menekan antara 21 hingga 30');

End;

**End.**

## 4.6. Case ... Of ... Else

Penambahan Else pada Case ... Of adalah untuk mengambil alih pilihan yang tidak terdapat pada daftar pilihan.

**Case variable of peluang; else bagian else; end;**

Bila pada program terdapat masukan di luar pilihan, secara otomatis proses program akan dilempar ke Else, bukan keluar dari struktur case seperti pada Case ... Of.

**Contoh Praktek 7 (Ketikkan dalam program Pascal)**

**Program Case\_03;**

**Uses wincrt;**

**Var**

**I : Integer;**

**Begin**

ClrScr;

Write ('masukkan suatu bilangan ');

Readln (I);

Case I of

1..10 :

Begin

Writeln ('Anda menekan antara 1 hingga 10');

Writeln ('masukan yang tepat !')

End;

11..20 :

Begin

Writeln ('Anda menekan antara 11 hingga 20');

Writeln ('Angka yang tertinggi adalah 20 !');

End;

21..30 :

Begin

Writeln ('Anda menekan antara 21 hingga 30');

Writeln ('Tepatnya adalah ',i);

End;

Else Begin

Writeln ('Masukan anda tidak memenuhi peluang ');

Writeln ('coba jalankan program sekali lagi');

End;

End;

**End.**

**Contoh Praktek 8 (Ketikkan dalam program Pascal)**

program case\_01;

Uses WinCrt;

Var

Angka : Real;

Pilihan : Integer;

Begin

```
ClrScr;  
Writeln('Konversi Ukuran Panjang ');  
writeln('Ketikkan angka yang anda pilih ');  
Writeln('1. Mile ke Kilometer');  
Writeln('2. Yard ke Meter');  
Writeln('3. Foot ke Meter');  
Writeln('4. Inch ke Centimeter');  
  
writeln;  
write('Pilihan : ');  
readln(pilihan);
```

## Case pilihan Of

1 : Begin

```
    write('Angka mile : ');readln(angka);  
    write(angka:5:3,' mile = ');  
    writeln(angka * 1.6093:10:4,' kilometer');  
end;
```

2 : Begin

```
    write('angka yard : ');readln(angka);  
    write(angka:5:3,' Yard = ');  
    writeln(angka * 0.9144:10:4,' meter');  
end;
```

3 : Begin

```
    write('Angka foot : ');readln(angka);  
    write(angka:5:3,' foot = ');  
    writeln(angka * 0.3048:10:4,' meter');  
end;
```

4 : Begin

```
    write('Angka inch : ');readln(angka);  
    write(angka:5:3,' mile = ');  
    writeln(angka * 2.54:10:4,' centimeter');  
end;
```

end;

end.



## Latihan

### Pilihan Ganda

1. Diberikan potongan algoritma sbb :

```
Input(N)  
If N >= 0 then  
    A ← -N  
End if  
Output(A)
```

Jika dari perintah Input(N), dimasukkan data -5, maka Tampilan dari Output(A) adalah

- |        |                |
|--------|----------------|
| a. 0   | d. 5           |
| b. -5  | e. salah semua |
| c. -10 |                |

2. Jika dari perintah Input(N) soal di atas, dimasukkan data 10, maka Tampilan dari Output(N) adalah

- |        |          |
|--------|----------|
| a. 0   | d. 5     |
| b. 10  | e. salah |
| c. -10 |          |
| semua  |          |

3. Diberikan potongan algoritma sbb :

```
Input(N)  
If N < 0 then  
    X ← -100  
Elseif N > 0 then  
    X ← 100  
Else  
    X ← 0  
End if  
Output(X)
```

Jika dari perintah Input(N), dimasukkan data 5, maka Output(X) menampilkan

- |      |                |         |
|------|----------------|---------|
| a. 0 | b. 100         | c. -100 |
| d. 5 | e. salah semua |         |

Perhatikan program berikut : (Soal nomor 4 – 6)

```
Program jika;  
uses WinCrt;  
Var  
    Suhu : Shortint;  
Begin  
    write ('Masukkan Nilai Suhu: ');  
    readln(Suhu);  
    If Suhu > 0 then  
        Write ('Bentuk Benda Beku')  
    Else If (Suhu>=0) And (Suhu<=100) then
```

```

Write('Bentuk Benda Cair')
Else
Write('Bentuk Benda Gas');
End.

```

4. Apabila Suhu dimasukkan nilai - 20 akan tampil
  - a. -20
  - b. Cair
  - c. Beku
  - d. Padat
  - e. Salah semua
5. Apabila Suhu dimasukkan nilai 20 akan tampil
  - a. -20
  - b. Cair
  - c. Beku
  - d. Padat
  - e. Salah semua
6. Apabila Suhu dimasukkan nilai 128 akan tampil
  - a. -20
  - b. Cair
  - c. Beku
  - d. Padat
  - e. Salah semua

**Perhatikan program berikut : (Soal nomor 7 - 10)**

```

program case_01;
Uses WinCrt;

Var
  Angka : Real;
  Pilihan : Integer;

Begin
  ClrScr;
  Writeln('Konversi Ukuran Panjang ');
  writeln('Ketikkan angka yang anda pilih ');
  Writeln('1. Mile ke Kilometer');
  Writeln('2. Yard ke Meter');
  Writeln('3. Foot ke Meter');
  Writeln('4. Inch ke Centimeter');

  writeln;
  write('Pilihan : ');
  readln(pilihan);

  Case pilihan Of

    1 : Begin
      write('Angka mile : ');readln(angka);

```

```

    write(angka:5:3,' mile = ');
    writeln(angka * 1.6093:10:4,' kilometer');
end;

```

**2 : Begin**

```

    write('angka yard : ');readln(angka);
    write(angka:5:3,' Yard = ');
    writeln(angka * 0.9144:10:4,' meter');
end;

```

**3 : Begin**

```

    write('Angka foot : ');readln(angka);
    write(angka:5:3,' foot = ');
    writeln(angka * 0.3048:10:4,' meter');
end;

```

**4 : Begin**

```

    write('Angka inch : ');readln(angka);
    write(angka:5:3,' Inch = ');
    writeln(angka * 2.54:10:4,' centimeter');
end;

```

**end;**

**end.**

7. Apabila Pilihan di input 2 dan angka di masukkan 2 akan tampil
  - a. 2 Mile = 1.8288 meter
  - b. 2 Yard = 2 meter
  - c. 2 Yard = 1.8288 meter
  - d. 2 Meter = 1.8288 Yard
  - e. Salah semua
8. Apabila Pilihan di input 3 dan angka di masukkan 200 akan tampil
  - a. 3 Foot = 200 meter
  - b. 3 Meter = 200 Foot
  - c. 3 Foot = 60.96 meter
  - d. 200 Meter = 3 foot
  - e. Salah semua
9. Apabila Pilihan di input 1 dan angka di masukkan 10 akan tampil
  - a. 1 Mile = 16.0930 Kilometer
  - b. 1 Mile = 10 Kilometer
  - c. 10 Mile = 1 Kilometer
  - d. 1 Mile = 1 Kilometer
  - e. Salah semua
10. Apabila Pilihan di input 4 dan angka di masukkan 10 akan tampil
  - a. 4 Inch = 10 Centimeter
  - b. 10 Inch = 25.4 Centimeter
  - c. 4 Inch = 1 Centimeter
  - d. 10 Inch = 10 Centimeter
  - e. Salah semua



## Essay

1. Ubahlah pernyataan-pernyataan berikut ini menjadi pernyataan IF .. THEN .. ELSE .. (Pergunakanlah identifier sesuai kreasi anda)
  - b. Jika honorarium lebih kecil dari 800000, kena pajak 2%
  - c. Jika potongan harga barang lebih mahal dari 15000000, mendapatkan potongan harga 4.5%.
  - d. Jika  $10 \leq X < 20$  maka nilai Y dihitung dari rumus  $Y = 2X^2 + 10 / X^3$ .
  - e. Jika  $X < 10$  atau  $X \geq 20$  maka nilai Y dihitung dari rumus  $Y = 10 * X - X^4 - 12$
2. Ubahlah pernyataan berikut ini menjadi pernyataan dengan CASE .. OF .. ELSE .. yang sesuai (pergunakan identifier sesuai kreasi anda)
  - a. Jika umurnya 16, maka kelas = 1  
Jika umurnya 17, maka kelas = 2  
Jika umurnya 18, maka kelas = 3
  - b. Jika namanya Agus, maka hobbynya berenang  
Jika namanya Boyke, maka hobbynya naik ke gunung  
Jika namanya Cherry maka hobbynya tari ballet  
Jika namanya Dianne, maka hobbynya balap mobil
  - c. Jika jaraknya 100 km, maka tariff Rp 60 / km  
Jika jaraknya 200 km, maka tariff Rp 70 / km  
Jika jaraknya 300 km, maka tariff Rp 80 / km  
Jika jaraknya 400 km, maka tariff Rp 90 / km  
Selain jarak yang disebutkan di atas, maka tariff rp 100 / km

3. Sebuah perusahaan mempekerjakan pegawai-pegawai dari golongan 1, 2, 3, 4, 5, 6. Peraturan gaji pada perusahaan itu menyatakan : Bila  $1 \leq \text{golongan} \leq 3$  maka gaji = (golongan \* jam kerja \* upah per jam) rupiah

Pajak = (2.5% dari gaji) rupiah

Libur = (21 / golongan) hari

Bonus = (golongan \* 25000) rupiah

Bila golongan = 4 atau 6 maka gaji = (golongan \* jam kerja \* upah per jam) rupiah.

Pajak = (6.5 % dari gaji) rupiah

Libur = (7 / golongan) hari

Bonus = (golongan \* 50000) rupiah

Bila golongan = 5 maka gaji = (golongan \* jam kerja \* upah per jam) rupiah

Pajak = (3.5% dari gaji) rupiah

Libur = (7 / golongan) hari

Bonus = (golongan \* 450000) rupiah

Buat program yang menanyakan golongan, jam kerja dan upah per jam, lalu menghitung besarnya gaji. Setelah dihitung, besarnya gaji, besarnya pajak, besarnya bonus, dan lamanya berlibur akan dicetak ke layar.

4. Peraturan penerimaan siswa pada suatu sekolah menyatakan bahwa :

Bila nilai rata-rata 7.5 s/d 10 maka dapat diterima, selain itu, dapat diterima asalkan membayar sumbangan sebesar (10/ nilai rata-rata \* 250000)

Buatlah program yang menanyakan nama dan nilai rata-rata calon siswa, lalu menentukan diterima tidaknya dan besarnya sumbangan yang harus dibayarkan.

5. Buatlah sebuah program yang menanyakan banyaknya anak dalam satu keluarga.

Jika banyaknya anak = 1 maka dicetak 'Keluarga sangat kecil'

Jika banyaknya anak = 2 maka dicetak 'Keluarga kecil'

Jika banyaknya anak = 3 maka dicetak 'keluarga sedang'

Jika banyaknya anak = 4 maka dicetak 'keluarga agak besar'

Jika banyaknya anak = 5 maka dicetak 'keluarga besar'.

Pergunakanlah pernyataan CASE..OF . untuk menyederhanakan program.

## BAB V PENGULANGAN PROSES

### 5.1. Perintah Goto

Untuk melaksanakan perintah Goto, diharuskan mendefinisikan suatu label terlebih dahulu. Label tersebut didefinisikan pada daerah deklarasi dengan kata label.

#### **Goto Label;**

#### **Contoh Praktek 27 (Ketikkan dalam program Pascal)**

##### **Program Luas1;**

##### **Uses WinCrt;**

Var

Lebar, tinggi, pajang, jari : Real;

Luas : real;

Pilihan : Integer;

Ch : Char

Label 10;

Begin

10 :

ClrScr;

Writeln ('Masukan tugas yang hendak dilakukan ');

Writeln ('1. menghitung luas segitiga')

Writeln ('2. Menghitung luas persegi panjang');

Writeln ('3. Menghitung luas Lingkaran');

Write ('Pilihan Anda :');

Readln (Pilihan);

Writeln;

Case Pilihan Of

1 : Begin

Write ('Masukan lebar alas :');

Readln (Lebar);

Write ('Masukan Tinggi Segitiga :');

Readln (tinggi);

Luas := 0.5 \* lebar \* tinggi;

Writeln ('Luas Segitiga adalah :',Luas:10:2);

End;

2 : Begin

Write ('Masukan lebar :');

Readln (Lebar);

Write ('Masukan panjang :');

Readln (panjang);

Luas := lebar \* panjang;

Writeln ('Luas Persegi empat adalah :',Luas:10:2);

End;

```

3 : Begin
    Write ('Masukan jari-jari :');
    Readln (jari);
    Luas := pi * sqr(jari);
    Writeln ('Luas lingkaran adalah :',Luas:10:2);
    End;
Else goto 10
    End;
Writeln;
Write ('Mau Melakukan perhitungan lagi ? (Y/N)');
Ch := Readkey;
If (Ch = 'Y') or (Ch = 'y') then goto 10;
End.

```

## 5.2. Perulangan For

For adalah perulangan dengan jumlah perulangan tertentu. Perulangan ini bergantung pada suatu variable integer. Pada setiap perulangan, variable tersebut akan bertambah atau berkurang satu.

**For V\_int := n\_awal To/ Downto n\_akhir do**  
**Proses yang dilaksanakan;**

### Contoh Praktek 28 (Ketikkan dalam program Pascal)

```

Program For1;
Uses wincrt;
Var
    I : Integer;
Begin
    Clrscr;
    For I := 1 to 10 do Writeln ('Nilai I adalah :',I);
End.

```

### Contoh Praktek 29 (Ketikkan dalam program Pascal)

```

Program For2;
Uses wincrt;
Var
    I : Integer;
Begin
    Clrscr;
    For I := 1 to 10 do
        Begin
            Writeln ('Nilai I adalah :',I);
            If I <> 10 then writeln ('Nilai I akan bertambah satu')
            Else Writeln ('Ini adalah nilai terakhir I');
        End;
    End;
End.

```



```

        End;
End.
Contoh Praktek 30 (Ketikkan dalam program Pascal)
Program For3;
Uses wincrt;
Var
    I : Integer;
Begin
    Clrscr;
    For I := 10 downto 1 do Writeln ('Nilai I adalah : ',I);
End.

```

### 5.3. Perulangan While..Do

Proses perulangan While..Do adalah proses dengan suatu kondisi. Kondisi yang dimaksud sama seperti pada if.. Jadi dapat disimpulkan bahwa While..Do dapat disamakan dengan suatu kondisi If yang dapat melakukan perulangan.

**While kondisi do perintah\_yang\_diulang;**

```

Contoh Praktek 31 (Ketikkan dalam program Pascal)
Program While1;
Uses WinCrt;
Var
    I : Integer;

Begin
    Clrscr;
    i := 1;
    While I <= 10 do
        Begin
            Writeln ('Nilai I adalah :',i);
            Inc(i)
        End;
End.

```

### **Contoh Praktek 32 (Ketikkan dalam program Pascal)**

```

Program While2;
Uses WinCrt;
Var
    I : Integer;
Begin
    Clrscr;
    i := 10;
    While (i >= 10) and ( i <= 20) do
        Begin

```

```
        Writeln ('Nilai I adalah :',i);  
        Inc(i)  
    End;  
End.
```

#### 5.4. Perulangan Repeat..Until

Perulangan Repeat..Until hampir sama dengan perulangan While..Do. perbedaannya adalah repeat..until mengecek kondisi pada akhir looping, sedang while..do pada awal looping.

##### **Repeat**

##### **Proses yang dilaksanakan**

##### **Until kondisi;**

#### **Contoh Praktek 33 (Ketikkan dalam program Pascal)**

```
Program repeat1;
Uses Wincrt;

Var
  I : Integer;

Begin
  Clrscr;
  i := 0;
  repeat
    inc(i);
    writeln ('Nilai I adalah :',i);
  Until i = 10;
End.
```

#### **Contoh Praktek 34 (Ketikkan dalam program Pascal)**

```
Program repeat2;
Uses Wincrt;

Var
  I : Integer;
Begin
  Clrscr;
  i := 0;
  repeat
    inc(i);
    j := 0
    repeat
      inc(j);
      writeln ('Nilai j adalah :',j);
    until j = 2;
    writeln ('Nilai I adalah :',i)
  Until i = 5;
End.
```

## LATIHAN

### PILIHAN GANDA

1. Penggalan Program sebagai berikut

```
Var
  h, i, j : Integer;
begin
  h:=0;
  for j := 1 to 10 do
    inc(h);i:=1;
    j := 0;
    repeat
      inc(i);
      inc(j);
    until i := 10;
    if h=j then writeln ('GOLD')
    else writeln ('SILVER');
  end.
```

Apa keluaran program di atas

SILVER

GOLD

- a. GOLD
- b. SILVER
- c. GOLD
- d. SILVER
- e. Semua salah

2. Begin

```
J := 1;
For i:= 1 to 5 do
Begin
  Writeln ( i , ' ' , j );
  j := i - 1;
end;
end.
```

Oputnya adalah .....

2 1  
1 3  
4 1  
1 5

a. 1 1

2 2  
3 3

b. 1 1

4 4  
5 5

c. 1 0

2 1  
3 2  
4 3  
5 4

d. 1 5

2 4  
3 3  
4 2  
5 1

e. 1 1

2 0  
3 1  
4 2  
5 3

3. Apakah output dari program berikut :

Var

i, j, k : integer

begin

j := 3;

k := 1;

Repeat

Dec(j);

For i:= 1 to j do

k:= 1 + 3 \* k

Dec(j);

Until j < 0;

Writeln (k);

End.

Outputnya adalah

a. 36

b. 46

c. 31

d. 13

e. Salah semua

4. Penggalan program dideklarasikan sebagai berikut :

Type ari = array[1..5] of byte;

Var

x :ari ;

i, j : byte;

```
.....  
for i:=1 to 5 do x [ i ]:=10 div i;  
j:=0; for i:=5 downto 1 do j:=j+x[ i ];  
writeln(j);
```

Output yang dicetak dari penggalan program di atas adalah :

- a. 5
- b. 22
- c. 22.8
- d. 5.22
- e. Salah semua

5. Masih berhubungan dengan soal nomor 4 di atas. Bila array di ubah dari 5 menjadi 10, begitu juga pada for i:=1 to 10 do dan for i:=10 downto 1 do maka, outputnya adalah...

- a. 28.3
- b. 28
- c. 27
- d. 26
- e. Salah semua

6. Penggalan program pascal sebagai berikut :

```
Y:=0;  
For i:=1 to 5 do  
  For j:=5 downto i do  
    Y:=Y+1;  
Writeln(Y);
```

Hasil outputnya :

- a. 15
- b. 20
- c. 35
- d. 40
- e. Salah semua

7. Masih berhubungan dengan nomor 6 di atas. Bila statemen Y:=Y+i; diganti dengan Y:=Y+j; maka outputnya :

- a. 55
- b. 45
- c. 35
- d. 25
- e. Salah semua

8. Masih menggunakan no. 6. Bila statemen Y:=Y+i, diganti dengan Y:=Y+1; maka outputnya menjadi

- a. 5
- b. 10

- c. 15
- d. 20
- e. Salah semua

9. Penggalan program sebagai berikut :

```
x:=0; y:=0;  
while x < 10 do  
begin  
    x:=x+1; y:=y+x;  
end;  
writeln(x, y);
```

Berapa harga x yang tercetak ?

- a. 9
- b. 10
- c. 11
- d. 12
- e. Salah semua

10. Masih berhubungan dengan no. 9, berapa harga y tercetak ?

- a. 55
- b. 45
- c. 36
- d. 35
- e. Salah semua

Penggalan program sebagai berikut :

```
A:=10; B:=0;  
Repeat  
    B:=B+1;  
    A:=A - B;  
Until A > B;  
Writeln(A, B);
```

11. Berapa harga A yang tercetak ?

- a. 1
- b. 10
- c. 9
- d. 2
- e. Salah semua

12. Masih berhubungan dengan nomor 11 harga B yang tercetak ?

- a. 1
- b. 10
- c. 9
- d. 2
- e. Salah semua

13. Penggalan program sebagai berikut :

```
A:=0; B:=10;
```

```
Repeat
```

```
  A:=A+1;
```

```
  B:=B - A;
```

```
Until A>B;
```

```
Writeln(A, B);
```

Berapa harga A yang tercetak ?

- a. 0
- b. 4
- c. 3
- d. 2
- e. Salah semua

14. Masih berhubungan dengan nomor 14, berapa harga B yang tercetak ?

- a. 4
- b. 3
- c. 0
- d. 2
- e. Salah semua

15. Penggalan program sebagai berikut :

```
For i:=1 to 3 do
```

```
  For j:=1 to i do
```

```
    Writeln('TOKI 97');
```

Berapa kali tercetak TOKI97?

- a. 9 kali
- b. 6 kali
- c. 3 kali
- d. 1 kali
- e. Salah semua

## Essay

1. Apakah perbedaan perulangan While..Do dengan Repeat..Until?



2. Buatlah sebuah program untuk menghitung factorial suatu bilangan. Input program adalah bilangan yang akan dicari faktorialnya, output program adalah nilai faktorialnya. Sebagai contoh, bila dimasukkan 5, keluaran program tersebut adalah :  $(1 * 2 * 3 * 4 * 5) 120 !$

3. Buatlah program yang dapat mencetak :

20 19 18 17 ... 3 2 1

dengan syarat :

Setiap selesai mencetak 5 buah bilangan maka dilayar akan dicetak pesan 'sudah lima bilangan'.

4. Buatlah sebuah program yang dapat menghitung rata-rata dua bilangan. Syaratnya adalah bilangan ke-1 harus lebih besar dari bilangan ke-2. Jika tidak memenuhi syarat, maka cetaklah komentar :

Maaf, syarat tidak dipenuhi.

Bilangan-1 harus lebih besar dari bilangan ke-2.

Kemudian mintalah lagi nilai bilangan ke-1 dan ke-2 sampai bilangan ke-1 lebih besar dari bilangan ke-2, baru hitung rata-ratanya.

5. Perhatikan deret bilangan berikut ini :

1,2,3,5,8,13,21,...

atau

3,4,7,11,18,...

atau

6,7,13,20,33,...

Deret bilangan seperti ini disebut bilangan FIBONACCI, rumus umumnya adalah :

$$Un = Un-1 + Un-2$$

Buatlah program yang dapat mencetak deretan bilangan FIBONACCI ke layar. Program diharapkan mempunyai urutan proses :

- menanyakan suku pertama dan suku kedua deret
- menanyakan banyaknya suku yang akan dicetak.
- Mencetak deret di layar.

6. Buatlah program yang menampilkan menu sebagai berikut :

Pilih salah satu :

L menghitung luas lingkaran

E menghitung luas segi empat persegi panjang

S menghitung luas segitiga

x keluar

Pilihan Anda =====>

Bila pilihannya bukan L, E, S atau x maka proses pemilihan akan diulang.

Kemudian program akan melakukan sesuai dengan pilihan.

7. Tampilkan deret bilangan FIBONACCI ke layar menggunakan :

- a. Loop REPEAT ... UNTIL ...
- b. Loop FOR ... TO ... DO

8. Buatlah program untuk mengetahui nilai factorial suatu bilangan. Gunakan pernyataan FOR .. TO .. DO ..

9. Buatlah program untuk mencetak bentuk seperti ini

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

\*\*\*\*\*

Dengan pernyataan REPEAT ... UNTIL ...

Dengan FOR .. TO .. DO ..

10. Buatlah program yang dapat melakukan proses sebagai berikut :

- Menanyakan 2 bilangan.
- Menanyakan operasi apa yang akan dilakukan terhadap 2 bilangan itu ( + , \* , / , - ).
- Mencetak hasilnya.

## **BAB VI PROSEDUR (bagian 1)**

### **6.1. Pendahuluan**

Selama ini, segala proses pemecahan masalah dilakukan di dalam bagian utama program, yaitu antara BEGIN dan END. Apabila masalah yang harus dipecahkan sangat besar dan kompleks, tentulah bagian utama program akan tampak sangat semrawut.

Satu-satunya cara untuk mengatasi hal ini adalah dengan menambahkan beberapa blok program baru ke dalam blok program yang sudah ada, untuk membantu bagian utama program.

Masalah yang sangat kompleks dipecah terlebih dahulu menjadi beberapa sub-masalah yang lebih sederhana. Setiap sub-masalah yang ada kemudian ditangani oleh sebuah blok\_program. Jadi, di dalam sebuah program Pascal akan terdapat bagian deklarasi, beberapa blok program pembantu, dan bagian utama program.

Blok program pembantu adalah program Pascal biasa yang ditulis menurut aturan tertentu. Jadi sebuah blok program akan memiliki judul, bagian deklarasi, dan kumpulan pernyataan diantara **BEGIN** dan **END**. Di dalam bahasa Pascal, blok program pembantu lebih dikenal dengan sebutan **MODUL PROGRAM**.

Bahasa Pascal mengenai adanya dua buah modul program, yaitu **prosedur (PROCEDURE)** dan **fungsi (FUNCTION)**, kedua jenis modul program ini merupakan ciri khas yang dimiliki bahasa Pascal.

Masalah yang akan dipecah-pecah menjadi beberapa sub-masalah. Setiap sub-masalah ditangani oleh sebuah prosedur atau fungsi tersendiri. Kumpulan beberapa prosedur dan fungsi akan membentuk program Pascal yang terpadu dan terstruktur.

Berbagai hal tentang prosedur dibahas dalam Bab 6 dan Bab 7, sedang tentang fungsi dibahas pada Bab 8.

### **6.2. Pengertian Prosedur**

Prosedur adalah modul program Pascal, yaitu block program Pascal yang digunakan untuk membantu bagian utama program.

**Contoh 6-1 :**

Program Mencari\_jumlah\_dan\_Ratarata;

Var

a, b , c : Integer;

Jumlah : Integer;

Rata : Real;

**Bagian deklarasi  
variabel**

Begin

{ ---- Penulisan Petunjuk --- }

    Writeln ('Apabila program ini dijalankan, ');

    Writeln ('Anda akan diminta menuliskan ');

    Writeln ('tiga buah angka. ');

    Writeln;

    Writeln ('Jumlah dan Rata\_2 dari ketiga ');

    Writeln ('bilangan tersebut akan di hitung');

    Writeln('dan ditampilkan ke layer ');

    Writeln;

**Bagian Utama  
Program**

{ --- menghitung jumlah dan rata\_rata --- }

    Write ('Masukkan tiga angka a, b, c : ');

    Readln (a, b, c);

    Jumlah := a + b + c;

    Rata := Jumlah/3;

    Writeln;

    Writeln ('Jumlahnya = ',Jumlah);

    Writeln ('Rata-ratanya = ',Rata);

End.

Perhatikan program pada contoh 6-1 di atas. Program tersebut memiliki dua pokok masalah, yaitu menampilkan petunjuk pemakaian program dan menghitung jumlah/ rata-rata tiga bilangan. Kedua masalah dicampur menjadi satu, yaitu dalam bagian program utama (diantara BEGIN dan END).

Agar program tampak terstruktur dengan baik, anda dapat memisahkan salah satu masalah menjadi sebuah blok program tersendiri.

Perhatian contoh program berikut :

### Contoh 6-2

Program Mencari\_jumlah\_dan\_Rata\_2;

Var

a, b , c : Integer;

Jumlah : Integer;

Rata : Real;

**Bagian deklarasi  
variabel**

PROCEDURE Petunjuk\_Program;

Begin

    Writeln ('Apabila program ini dijalankan, ');

    Writeln ('Anda akan diminta menuliskan ');

    Writeln ('tiga buah angka.

    Writeln;

    Writeln ('Jumlah dan Rata\_2 dari ketiga ');

    Writeln ('bilangan tersebut akan di hitung');

    Writeln('dan ditampilkan ke layer ');

    Writeln;

End;

**Blok Prosedur  
Petunjuk\_Program**

Begin

    Petunjuk\_Program;

    Write ('Masukkan tiga angka a, b, c : ');

    Readln (a, b, c);

    Jumlah := a + b + c;

    Rata := Jumlah/3;

    Writeln;

    Writeln ('Jumlahnya = ',Jumlah);

    Writeln ('Rata-ratanya = ',Rata);

End.

**Bagian Utama  
Program**

Pada program contoh 6-2, bagian program untuk menampilkan petunjuk yang semula berada di dalam bagian utama program dipisah menjadi sebuah blok program tersendiri, Blok program ini disebut PROCEDURE. Di dalam bahasa pemrograman lain, procedure dikenal sebagai SUBROUTIN.

Perhatikan bahwa program pada contoh 6-2 terdiri dari sebuah bagian deklarasi variable, sebuah blok prosedur, dan bagian utama program. Blok prosedur dapat dipanggil dari bagian utama program dengan menyebut namanya.

### 6.3. Deklarasi Prosedur

Prosedur memiliki struktur yang sama dengan program Pascal yang anda kenal selama ini. Prosedur harus memiliki judul serta pernyataan yang diapit oleh BEGIN dan END. Selain itu, sebuah prosedur boleh memiliki bagian deklarasi CONST, VAR, dan TYPE sendiri.

Judul sebuah prosedur harus dimulai dengan kata PROCEDURE yang diikuti oleh nama prosedur tersebut, dan diakhiri oleh tanda titik koma (;). Nama prosedur terdiri dari maksimal 63 huruf, tidak boleh mengandung spasi, dan harus diawali dengan huruf (alphabet). Selain itu, nama suatu prosedur tidak boleh sama dengan nama variable yang digunakan oleh program.

Contoh 6-3.1 :

- Judul prosedur yang benar :

PROCEDURE Petunjuk\_program;

PROCEDURE Maksimum;

PROCEDURE Cari\_jumlah;

- Judul Prosedur yang salah :

PROCEDURE Petunjuk Program (karena mengandung spasi)

PROCEDURE 123BL (Karena diawali angka)

Perhatikan kembali blok prosedur Petunjuk\_Program pada contoh 6-2. kumpulan pernyataan yang terdapat di dalam prosedur tersebut diawali oleh BEGIN dan diakhiri oleh END. Kata END yang mengakhiri suatu prosedur tidak diikuti oleh tanda titik, melainkan tanda titik koma (;). Tanda titik (.) hanya diperbolehkan mengikuti kata END milik bagian utama program.

Prosedur tidak dapat berdiri sendiri, melainkan harus di dalam sebuah program Pascal. Di dalam setiap program Pascal, blok prosedur harus diletakan sesudah bagian deklarasi, tetapi sebelum bagian utama program.

Prosedur yang telah dibuat dapat dipanggil oleh bagian utama program dengan cara menyebut nama prosedur yang bersangkutan. Pada saat suatu prosedur dipanggil, aliran program akan meloncat ke awal blok prosedur tersebut, lalu mengerjakan semua pernyataan yang ada disana. Setelah selesai, aliran program akan kembali ke tempat semula, yaitu di dalam bagian utama program.



### Contoh 6-3.2

#### **Program untuk menghitung luas bola dengan memakai prosedur**

PROGRAM Menghitung\_Luas\_Bola;

Const

pi = 3.14;

Var

Jari : Integer;

Luas : Real;

**Bagian deklarasi  
konstanta dan variabel**

PROCEDURE Hitung\_Luas;

BEGIN

Luas := 4\*pi\*sqr(Jari);

Writeln('Luas Bola = ', Luas);

END;

**Blok prosedur  
Hitung Luas**

BEGIN

Write ('tuliskan jari-jari Bola = ');

Readln(Jari);

Hitung\_luas;

Writeln(' --- selesai --- ');

END.

**Bagian  
Utama  
Program**

Jari-jari bola ditanyakan pada bagian utama program dan disimpan ke dalam variable jari. Setelah itu prosedur Hitung\_Luas dipanggil. Pada saat prosedur Hitung\_luas dipanggil, aliran program akan meloncat ke awal prosedur Hitung\_luas, lalu mengerjakan program yang ada di dalamnya (yaitu menghitung luas bola). Setelah selesai, aliran program meloncat kembali ke bagian utama program, dan mencetak kata-kata --- selesai ---.

#### **6.4. Ruang Lingkup Konstanta, Tipe dan Variabel**

Seperti halnya program Pascal yang sudah anda kenal, prosedur boleh memiliki bagian deklarasi sendiri. Bagian deklarasi di dalam prosedur meliputi deklarasi konstanta, tipe, maupun variable.

Deklarasi yang dilakukan di dalam suatu blok prosedur disebut DEKLARASI LOKAL. Konstanta, tipe maupun variable yang dideklarasikan

secara local hanya akan dikenali oleh blok procedure yang bersangkutan. Bagian program di luar blok prosedur tidak mengenali konstanta, tipe, dan variable yang dideklarasikan secara local.

Deklarasi yang dilakukan diluar blok procedure, tetapi masih di dalam tubuh program utama disebut DEKLARASI GLOBAL. Semua konstanta, tipe, dan variable yang dideklarasikan secara global akan dikenali oleh seluruh tubuh program, termasuk blok-blok prosedur yang ada.

Contoh 6-4.1

**Perhatikan Program Pada contoh 6.3.2 yang ditulis kembali sebagai berikut :**

```
PROGRAM Menghitung_Luas_Bola;
```

```
Const
```

```
    pi = 3.14;
```

```
Var
```

```
    Jari : Integer;
```

**Bagian deklarasi  
Global**

```
PROCEDURE Hitung_Luas;
```

```
Var
```

```
    Luas : Real;
```

**Deklarasi  
Lokal**

**Blok prosedur  
Hitung Luas**

```
BEGIN
```

```
    Luas := 4*pi*sqr(Jari);
```

```
    Writeln('Luas Bola = ', Luas);
```

```
END;
```

```
BEGIN
```

```
    Write ('tuliskan jari-jari Bola = ');
```

```
    Readln(Jari);
```

```
    Hitung_Luas;
```

```
    Writeln(' --- selesai --- ');
```

```
END.
```

**Bagian  
Utama  
Program**

Perhatikan letak deklarasi global dan deklarasi local pada program di atas. Deklarasi global diletakkan di luar prosedur, tetapi masih di dalam

tubuh program. Deklarasi local diletakkan di dalam prosedur Hitung\_luas.

## Contoh Soal

1. Perhatikan Program berikut,

```
PROGRAM Menulis_Kesan;
```

```
PROCEDURE Kesan;
```

```
BEGIN
```

```
  Writeln ('saya sedang belajar bahasa Pascal ');
```

```
  Writeln ('saya pusing pisan ');
```

```
  Writeln ('walaupun saya sudah berusaha tetap aja bingung....');
```

```
END;
```

```
Begin
```

```
  Kesan;
```

```
  Writeln (' dan kepala saya pusing tujuh keliling....');
```

```
End.
```

f. Tuliskan dan jalankan program di atas !

g. Jelaskan cara kerja program !

Jawab

a. ....

b. Program di atas terdiri dari sebuah prosedur, yaitu prosedur kesan. Prosedur ini dipanggil dari bagian utama program. Pada saat prosedur kesan dipanggil, aliran program akan meloncat ke awal prosedur tersebut, dan menulis kalimat :

Saya sedang belajar bahasa pascal

Saya pusing pisan

Walaupun saya sudah berusaha tetap aja bingung...

Setelah selesai, aliran program kembali ke bagian utama program lalu menulis kalimat :

Dan kepala saya pusing tujuh keliling

2. Perhatikan Program berikut,

```
Program Menampilkan_Pesan;
```

```
Var
```

```
  A : String[20];
```

```
PROCEDURE Satu;
```

```
BEGIN
```

```
  Writeln ('saya sedang belajar bahasa pascal ');
```

END;

```

PROCEDURE Dua;
BEGIN
    Writeln ('saya pusing pisan');
END;

```

```

PROCEDURE Tiga;
BEGIN
    Writeln ('walaupun sudah berusaha ...');
END;

```

```

BEGIN
    Write ('Tulis nama anda : ');
    Readln (a);
    Writeln (' Nama Saya : ', a);
    Satu;
    Dua;
    Tiga;
END;

```

- b. Tuliskan dan jalankan program di atas !
- c. Sebutkan semua variable yang digunakan serta tipe dan ruang lingkupnya !
- d. Jelaskan aliran kerja program !

Jawaban

- a. ....
- b. a adalah variable jenis string yang memiliki ruang lingkup global, sehingga dikenal oleh seluruh bagian program.
- c. Blok program utama akan menanyakan nama dan menyimpannya di variable a, setelah itu, prosedur satu dipanggil, disusul prosedur dua dan terakhir prosedur tiga. Perhatikan cara memanggil prosedur (dengan menuliskan nama prosedur\_tersebut).

3. Perhatikan Program berikut,  
Program Luas\_dan\_Volume\_Bola;  
Const

Pi = 3.14;

Var

r : Integer;

luas : real;

isi : real;

PROCEDURE Tanya\_jari\_jaril;

BEGIN

Write ('Tuliskan jari-jari Bola : ');

Readln (r);

END;

PROCEDURE Hitung\_luas;

BEGIN

Luas := 4 \* pi \* r \* r

```

END;

PROCEDURE Hitung_volume;
BEGIN
    Isi := 4/3 * pi * r * r * r
END;
BEGIN
    Tanya_jari_jari;
    Hitung_luas;
    Hitung_Volume;
    Writeln ('Luas Bola = ', luas);
    Writeln ('Volume Bola = ', Isi);
END.

```

- (a) Ketikkan dan jalankan program di atas !
- (b) Jelaskan aliran kerja program diatas !
- (c) Sebutkan semua variable yang digunakan, beserta ruang lingkupnya !
- (d) Dapatkah prosedur Tanya\_jari\_jari dipanggil dari prosedur hitung\_luas ?
- (e) Dapatkah prosedur Hitung\_luas dipanggil dari Tanya\_jari\_jari ?

Jawab

- a. ....
- b. Program di atas terdiri dari sebuah blok deklarasi konstanta, sebuah deklarasi variable, tiga buah prosedur dan sebuah blok program utama. Prosedur Tanya\_jari\_jari akan menanyakan jari-jari lingkaran, prosedur hitung\_luas menghitung luas bola, dan prosedur hitung\_volume menghitung volume bola. Ketiga prosedur dipanggil dari blok program utama. Mula-mula prosedur Tanya\_jari\_jari dipanggil. Setelah variabel terisi, maka prosedur hitung\_luas dipanggil. Luas bola dihitung oleh prosedur ini dan hasilnya disimpan ke dalam variable luas. Selanjutnya prosedur hitung\_volue dipanggil. Volume bola dihitung oleh prosedur ini, dan hasilnya ke dalam variable isi.
- c. pi adalah konstanta global, r, Luas dan isi adalah variable global. Semua konstanta dan variable tersebut dikenali oleh seluruh bagian program.
- d. Dapat, sebab prosedur Tanya\_jari\_jari diletakkan sebelum prosedur hitung\_luas.
- e. Tidak dapat, sebab prosedur hitung\_luas diletakkan di bawah prosedur Tanya\_jari\_jari.

## Latihan

1. Perhatikan Program di bawah ini :

```
Program luas_dan_volume_bola;  
PROCEDURE Hitung_luas_dan_volume;  
Const  
    pi = 3.14;  
var  
    r : Integer;  
    Luas, Isi : Real;
```

```
BEGIN
```

```
    Write ('Tuliskan jari jari Bola : ');  
    Readln (r);  
    Luas := 4 * pi * r * r;  
    Isi := 4/3 * pi * r * r * r;  
    Writeln (' luas Bola = ', Luas);  
    Writeln ('Volume Bola = ', Isi);
```

```
END;
```

```
BEGIN
```

```
    Hitung_luas_dan_Volume;
```

```
END.
```

- i. Ketikkan dan jalankan program di atas !
- ii. Jelaskan aliran kerja program di atas !
- iii. Sebutkan semua variable yang digunakan beserta ruang lingkupnya !

2. Perhatikan Program Berikut :

```
Program Mengubah_satuan;
```

```
Const
```

```
    AkhirData=0;  
    Faktor=100;
```

```
Var
```

```
    Meter : Real;  
    Sentimeter:real;
```

```
PROCEDURE Petunjuk;
```

```
Var
```

```
    Jawab : Char;
```

```
BEGIN
```

```
    Write ('Mau melihat petunjuk (Y/ N) : ');  
    Readln (Jawab);  
    If Jawab = 'Y' then
```



```

Begin
    Writeln ( ' Program ini digunakan untuk ');
    Writeln ( ' mengubah panjang dalam satuan ');
    Writeln ( ' meter menjadi sentimeter ');
    Write);
    Writeln ( ' Setiap kali anda akan ditanya ');
    Writeln ( ' panjang dalam satuan meter, lalu ');
    Writeln ( ' program akan mencetaknya dalam ');
    Writeln ( ' satuan sentimeter ');
    Writeln);
    Writeln ( ' Bila anda ingin mengakhiri program ');
    Writeln ( ' ini, masukkan angka 0 ');
End;
End;

```

BEGIN

Petunjuk;

Repeat

Writeln;

Write('Tuliskan panjang (dalam satuan meter) : ');

Readln(Meter);

If Meter <> AKhirData Then

Begin

Sentimeter := Faktor \* Meter;

Writeln('Panjang = ', sentimeter, ' cm');

End;

Until Meter = AKhirData;

Writeln;

Writeln(' --- program berakhir ---');

End.

Pertanyaan

- ii. Tuliskan dan jalankan program di atas !
- iii. Sebutkan semua variable dan konstanta yang digunakan beserta tipe dan ruang lingkupnya !
- iv. Jelaskan aliran kerja program !

3. Perhatikan program berikut :

```
PROGRAM Memasukkan_Bilangan;
```

```
    PROCEDURE negatif;
```

```
    BEGIN
```

```
        Writeln ('Bilangan yang anda ketikkan negatif')
```

```
    END;
```

```
    PROCEDURE positif;
```

```
    BEGIN
```

```
        Writeln (' Bilangan yang anda ketikkan positif ');
```

```
    END;
```

```
    PROCEDURE Nol;
```

```
    BEGIN
```

```
        Writeln (' Bilangan yang anda ketikkan nol ');
```

```
    END;
```

```
    PROCEDURE input;
```

```
    VAR
```

```
        Angka : Integer;
```

```
    BEGIN
```

```
        Writeln (' Masukkan sembarang angka : ');
```

```
        Readln (Angka);
```

```
        IF Angka > 0 Then Positif;
```

```
        IF Angka = 0 Then Nol;
```

```
        IF ANGka < 0 Then Negatrif;
```

```
    END;
```

```
BEGIN
```

```
    Input;
```

```
END.
```

1. Ketikkan dan jalankan program di atas
2. Jelaskan aliran kerja program !

4. Buatlah program yang apabila dijalankan akan menanyakan suhu dalam derajat Celcius, kemudian menampilkan kembali suhu tersebut dalam derajat Fahrenheit, derajat Reamur, dan derajat Kelvin. Gunakan paling sedikit 3 buah prosedur.

5. Buatlah program untuk menghitung akar-akar suatu persamaan  $Ax^2 + Bx + C$ . harga konstanta, A, B, C ditanyakan terlebih dahulu, lalu ketiganya dipakai untuk menghitung diskriminan D.

Jika  $D > 0$  maka terdapat dua akar riil yang berbeda

Jika  $D = 0$  maka terdapat dua akar riil yang kembar

Jika  $D < 0$  maka akar-akarnya tidak riil

6. Sebuah benda bermassa  $M$  Kg jatuh bebas dari ketinggian  $H$  meter.

Buatlah program untuk menghitung :

- a) Waktu yang dibutuhkan untuk mencapai tujuan
- b) Kecepatan saat peluru mencapai tanah.

## BAB VII PROSEDUR (Bagian 2)

### 7.1. Prosedur Berganda

Apabila diperlukan, kalian dapat memanggil suatu prosedur pada saat anda sedang berada di dalam prosedur yang lain. Tentu saja, prosedur yang dipanggil tersebut harus sudah dibuat dahulu sebelumnya.

**Contoh 7-1** perhatikan program berikut

Program Jumlah\_dan\_Hasil\_Kali;

Var

a, b, c : Integer;

Jumlah : Integer;

Kali : Integer;

PROCEDURE Penjumlahan;

Begin

Jumlah := a + b + c;

End;

Blok  
Prosedur  
Pertama

PROCEDURE Perkalian;

Begin

Kali := a \* b \* c;

End;

Blok  
Prosedur  
Kedua

PROCEDURE Jumlah\_Kali;

Begin

Penjumlahan;

Perkalian;

Writeln ('jumlahnya = ', jumlah);

Writeln ('Perkaliannya = ', kali);

End;

Blok  
Prosedur  
Ketiga

<pre> Begin     Write ('Tuliskan tiga buah bilangan : ');     Readln (a, b, c);     Jumlah_kali; End; </pre>	<div style="border-left: 1px solid black; height: 100px; margin-left: 10px;"></div>	Bagian Utama Program
--	---	----------------------------

Blok prosedur pertama diberi nama Penjumlahan, blok prosedur kedua diberi nama Perkalian, dan blok prosedur ketiga diberi nama Jumlah\_Kali;

Variable jumlah dan kali dideklarasikan secara global, agar dapat dikenali oleh seluruh tubuh program termasuk prosedur perkalian, penjumlahan, dan jumlah\_kali;

Pada saat program dijalankan, bagian utama program akan memanggil prosedur jumlah\_kali dideklarasikan secara terpisah. Prosedur, penjumlahan dideklarasikan lebih dahulu, disusul prosedur perkalian, dan terakhir prosedur jumlah\_kali.

Suatu prosedur hanya dapat dipanggil apabila prosedur tersebut telah dideklarasikan terlebih dahulu sebelumnya. Prosedur jumlah\_kali dapat memanggil prosedur penjumlahan dan prosedur perkalian, sebab prosedur penjumlahan dan perkalian dideklarasikan di atas prosedur jumlah\_kali.

Apabila dikehendaki, prosedur perkalian dapat memanggil prosedur penjumlahan. Akan tetapi prosedur penjumlahan tidak dapat memanggil prosedur perkalian, sebab prosedur perkalian dideklarasikan di bawah prosedur penjumlahan.

Prosedur jumlah\_kali tidak dapat dipanggil oleh prosedur penjumlahan dan perkalian. Prosedur jumlah\_kali hanya dapat dipanggil oleh bagian utama program.

Kalian dapat mendeklarasikan suatu prosedur di dalam prosedur lain. Dalam keadaan seperti ini, prosedur yang dideklarasikan tersebut hanya akan dikenali oleh blok prosedur tempat pendeklarsian berlangsung.

## Contoh 7-2

Program Jumlah\_dan\_Hasil\_Kali;

Var

a, b, c : Integer

PROCEDURE jumlah\_kali;

Var

Jumlah : Integer;

Kali : Integer;

PROCEDURE Penjumlahan;

Begin

Jumlah := a + b + c;

End;

Blok  
Prosedur  
penjumlahan

PROCEDURE Perkalian;

Begin

Kali := a \* b \* c;

End;

Blok  
Prosedur  
perkalian

Begin

Penjumlahan;

Perkalian;

Writeln ('jumlahnya = ', jumlah);

Writeln ('Perkaliannya = ', kali);

End;

Blok  
Prosedur  
Jumlah\_kali

Begin

Write ('Tuliskan tiga buah bilangan : ');

Readln (a, b, c);

Jumlah\_kali;

End;

Bagian  
Utama  
Program

Perhatikan bahwa prosedur Penjumlahan dan Perkalian terletak di dalam prosedur Jumlah\_kali. Oleh sebab itu, prosedur penjumlahan dan prosedur perkalian disebut memiliki ruang lingkup local terhadap prosedur jumlah\_kali.

Prosedur\_penjumlahan dan perkalian hanya dikenali oleh blok prosedur jumlah\_kali, sehingga keduanya hanya dapat dipanggil dari dalam blok prosedur jumlah\_kali. Bagian utama program tidak mengenali prosedur

penjumlahan dan perkalian, sehingga kedua prosedur tersebut tidak dapat dipanggil oleh bagian utama program.

Prosedur penjumlahan dan prosedur perkalian hanya akan dijalankan apabila prosedur jumlah\_kali dijalankan.

Kesimpulan yang dapat diperoleh adalah bahwa suatu prosedur hanya dapat dipanggil dari dalam blok tempat prosedur tersebut berada.

## **7.2. Parameter Nilai dan Parameter Variabel**

Pada proses pemanggilan prosedur, terjadi pertukaran data antara blok prosedur dengan blok pemanggil. Perhatikan program pada contoh 7-1 di atas. Variable a, b, dan c diisi dengan data melalui pernyataan readln di dalam bagian utama program. Data yang tersimpan di dalam variable a, b, dan c. ini selanjutnya akan dioleh oleh prosedur jumlah\_kali. Dengan kata lain, pertukaran data antara blok pemanggil (dalam hal ini adalah bagian utama program) dengan blok prosedur terjadi melalui variable a, b, dan c.

Untuk keperluan pertukaran data ini, tentu saja variable a, b, dan c harus dideklarasikan sebagai variable global, agar dikenali oleh kedua blok (blok prosedur dan bagian utama program).

Variable global bukan media yang baik untuk melakukan pertukaran data, sebab penggunaan variable global untuk keperluan ini akan mengubah isi variable global tersebut setiap kali prosedur dijalankan.

Cara terbaik untuk mengatasi hal ini adalah dengan menggunakan PARAMETER. Parameter berfungsi sebagai 'media perantara' antara variable di dalam blok prosedur dengan variable di dalam blok pemanggil. Pada modul ini, kepada Anda hanya akan diperkenalkan dua jenis parameter, yaitu PARAMETER NILAI (Value Parameter) dan PARAMETER VARIABEL (Variabel Parameter).

Pada saat suatu prosedur dipanggil, isi variable-variabel yang terdapat di dalam blok pemanggil akan 'disalin' ke dalam parameter-parameter yang tersedia di dalam blok prosedur. Apabila parameter yang dipakai di dalam blok prosedur adalah parameter nilai, maka proses yang

terjadi di dalam blok prosedur hanya melibatkan parameter-parameter milik prosedur tersebut. Isi variable yang terdapat di dalam blok pemanggil tidak akan berubah.

Akan tetapi apabila parameter yang digunakan di dalam blok prosedur adalah parameter variable, maka proses yang terjadi di dalam prosedur akan mempengaruhi isi variable yang terdapat di dalam blok pemanggil.



### Contoh 7-3

```
Program Jumlah_dan_Hasil_Kali;  
Var  
    a, b, c : Integer;  
  
PROCEDURE jumlah_kali(x, y : Integer);  
Var  
    Jumlah, kali : Integer;  
  
Begin  
    Jumlah := x + y + z;  
    Kali := x * y * z;  
    Writeln('jumlahnya = ', jumlah);  
    Writeln('perkaliannya = ', kali);  
End;  
Begin  
    Write ('tuliskan tiga bilangan :');  
    Readln(a, b, c);  
    Jumlah_kali(a, b, c);  
End.
```

Nama-nama parameter (beserta jenisnya) yang akan dipergunakan oleh prosedur harus dituliskan pada judul prosedur (di dalam tanda kurung, setelah nama prosedur).

Pada program contoh 7-3 diatas, x, y, dan z adalah parameter Nilai jenis integer yang terdapat di dalam prosedur jumlah\_kali, sedangkan a, b, dan c adalah variable global bertipe integer yang terdapat di dalam blok program utama.

Perhatikan bahwa seluruh parameter yang akan dipakai oleh blok prosedur beserta tipenya harus ditulis/ dinyatakan di dalam judul prosedur, dan diapit oleh tanda kurung (x, y, z : Integer).

Perhatikan bahwa seluruh parameter yang akan dipakai oleh blok prosedur beserta tipenya harus ditulis/ dinyatakan di dalam judul prosedur, dan diapit oleh tanda kurung (x, y, z : Integer).

Perhatikan pula cara memanggil prosedur jumlah\_kali, yaitu dengan menuliskan namanya diikuti deretan variable yang akan digunakan oleh blok prosedur (a, b, dan c).

Pada saat prosedur jumlah\_kali dipanggil, isi variable a akan disalin ke x, b, ke y, dan c ke z. pada saat blok prosedur jumlah\_kali dijalankan, yang terlibat hanyalah parameter integer x, y dan z. isi variable a, b dan c tidak akan berubah. Walaupun isi parameter nilai x, y dan z berubah.

#### **Contoh 7-4 Perhatikan program berikut :**

```
Program Jumlah_dan_Hasil_Kali;  
Var  
    a, b, c : Integer;  
PROCEDURE jumlah_kali(Var x, y, z : Integer );  
Var  
    Jumlah, kali : Integer;  
Begin  
    Jumlah := x + y + z;  
    Kali := x * y * z;  
    Writeln ('jumlahnya = ', jumlah);  
    Writeln ('perkaliannya = ',kali);  
End;  
  
Begin  
    Write ('tuliskan tiga bilangan : ');  
    Readln (a, b, c);  
    jumlah_kali(a, b, c);  
End.
```

Pada program contoh 7-4 di atas, x, y dan z adalah Parameter Variabel bertipe integer yang terdapat di dalam prosedur jumlah\_kali. Perhatikan kata VAR di dalam judul prosedur. Kata VAR ini menandakan bahwa x, y dan z adalah parameter variable (bedakan dengan cara menyatakan Parameter Nilai).

Pada saat prosedur jumlah\_kali dipanggil, isi variable a akan disalin ke x, b ke y, dan c ke z. pada saat blok prosedur jumlah\_kali dijalankan, apabila isi parameter x berubah, maka isi variable global a juga berubah. Bila isi y berubah, b juga berubah, demikian pula apabila z berubah, c juga berubah. Ada beberapa hal yang harus anda perhatikan pada waktu menggunakan parameter.

- b. Banyaknya variable yang disebutkan pada waktu memanggil prosedur harus sama dengan banyaknya parameter yang tertulis pada judul prosedur. Pada contoh 7-4 di atas, terdapat tiga variable global a, b dan c. Pada judul prosedur juga terdapat tiga buah parameter, yaitu x, y dan z.
- c. Tipe variable yang disebutkan pada waktu memanggil prosedur harus sama dengan tipe parameter yang tertulis di dalam judul

prosedur. Pada contoh 7.4 di atas, variable a,b dan c memiliki tipe integer. Demikian pula x, y dan z.

## Soal

### 1. Perhatikan Program Berikut

```
PROGRAM Luas_dan_Volume_Bola;

    PROCEDURE Hitung_luas_dan_Volume;
    CONST
        pi = 3.14

    VAR
        r : Integer;
        Luas : real;
        Isi : Real;

    PROCEDURE Tanya_Jari_Jari;
    BEGIN
        Write ('Tuliskan Jari-jari Bola : ');
        Readln(r);
    END;

    PROCEDURE Hitung_luas;
    BEGIN
        Luas := 4 * pi * r * r
    END;

    PROCEDURE Hitung_Volume;
    BEGIN
        Isi := 4/3 * pi * r * r * r
    END;
```

### **BEGIN**

```
    Hitung_Luas_dan_Volume;
```

### **END.**

Pertanyaan :

- Ketikkan dan jalankan program di atas !
- Jelaskan aliran kerja program di atas !
- Sebutkan semua variable dan konstanta yang digunakan beserta ruang lingkupnya !
- Dapatkah prosedur Tanya\_jari\_jari dipanggil dari blok program utama ?

- e. Dapatkah prosedur Hitung\_luas memanggil prosedur Tanya\_jari\_jari ?
- f. Dapatkah prosedur Hitung\_luas memanggil prosedur hitung\_volume ?

Jawab

- a. ...
- b. Program di atas terdiri dari bagian utama program dan sebuah prosedur (yaitu prosedur hitung\_luas\_dan\_volume). Di dalam prosedur Hitung\_luas\_dan\_volume terdapat tiga buah prosedur local, yaitu prosedur Tanya\_jari\_jari, prosedur Hitung\_luas, dan prosedur Hitung\_volume ketiga prosedur tersebut, hanya dikenali oleh blok prosedur. Hitung\_luas\_dan\_volume sehingga anda hanya dapat memanggilnya dari blok prosedur hitung\_luas\_dan\_volume. Pada saat prosedur hitung\_luas\_dan\_volume dipanggil, perjalanan program akan meloncat ke awal prosedur tersebut, selanjutnya prosedur Tanya\_jari\_jari dipanggil, disusul oleh prosedur hitung\_luas dan terakhir prosedur hitung\_volume, setelah semua prosedur dipanggil, hasil perhitungan ditampilkan di layar.
- c. Konstanta pi, variable r (integer), variable luas (real) dan variable isi (real) dideklarasikan secara local terhadap prosedur hitung\_luas\_dan\_volume.
- d. Tidak dapat sebab prosedur Tanya\_jari\_jari adalah prosedur local yang hanya dikenali oleh blok prosedur hitung\_luas\_dan\_volume.
- e. Dapat, sebab kedua prosedur berada di dalam blok yang sama (yaitu blok prosedur hitung\_luas\_dan\_volume). Alasan lain adalah karena prosedur Tanya\_jari\_jari diletakkan sebelum prosedur hitung\_luas.
- f. Tidak dapat, sebab walaupun keduanya berada di dalam blok yang sama, prosedur Hitung Volume diletakkan sesudah prosedur hitung\_luas.

## 2. Perhatikan Program di bawah ini !

```
PROGRAM Luas_dan_Volume_Bola;
```

```
    PROCEDURE Hitung_luas_dan_Volume;
```

```
    CONST
```

```
        pi = 3.14;
```

```
    VAR
```

```
        R : Integer;
```

Luas, Isi : Real;

BEGIN

Write ('Tuliskan Jari-jari Bola : ');

Readln (r);

Luas := 4 \* pi \* r \* r;

Isi := 4/3 \* pi \* r \* r \* r;

Writeln ('Luas Bola = ', luas);

Writeln ('Volume Bola = ', Isi);

END;

BEGIN

Hitung\_luas\_dan\_Volume;

END.

Pertanyaan

- Ketikkan dan jalankan program di atas !
- Apakah perbedaan yang antara program di atas dengan program pada soal 1.

### 3. Perhatikan program berikut :

PROGRAM Luas\_dan\_volume\_bola;

VAR

Jari : Integer;

PROCEDURE Hitung\_luas\_dan\_Volume (r : Integer);

CONST

pi = 3.14;

VAR

Luas, Isi : Real;

BEGIN

Luas := 4 \* pi \* r \* r;

Isi := 4/3 \* pi \* r \* r;

Writeln (' Luas Bola = ', Luas);

Writeln (' Volume Bola = ', Isi);

END;

BEGIN

Write ('Tuliskan Jari-Jari Bola : ');

Readln (Jari);

Hitung\_luas\_dan\_volume(jari);

END.

Soal

- Ketikkan dan jalankan program di atas !
- Jelaskan aliran kerja program di atas !



- c. Sebutkan semua variable yang digunakan beserta ruang lingkupnya.
- d. Apakah perbedaan program di atas dengan program pada soal nomor 2.

#### 4. Perhatikan Program Berikut :

```
PROGRAM Memasukkan_Angka;

VAR
    Angka : Integer;

    PROCEDURE Input;
    BEGIN
        Write ('Ketikkan sembarang angka positif : ');
        Readln (Angka);
        IF (Angka < 0) Then Input;
    END;

BEGIN
    Input;
    Writeln ('Anda mengetikkan angka dengan benar');
    Writeln ('Angka yang anda ketikkan = ', angka);
END.
```

Soal

- a. Ketikkan dan jalankan program di atas !
- b. Jelaskan aliran kerja program di atas !

#### 5. Perhatikan program berikut

```
Program Menghitung_faktorial;

Var
    Bil : Integer;

    PROCEDURE Faktorial (n : Integer);
    Var
        Hasil : Integer;
        Angka : Integer;
    BEGIN
        Hasil :=1;
        Angka :=n;
        While n>0 DO
        BEGIN
            Hasil := Hasil * n;
            n := n - 1
        
```

```

        END;
        Writeln (Angka, ' factorial = ', hasil)
    END;
BEGIN
    Write ('Masukkan bilangan bulat positif :');
    Readn (Bil);
    Faktorial (Bil);
    Writeln ('Bilangan yang anda masukkan = ',Bil)
END.

```

Soal

- Ketikkan dan jalankan program di atas.
- Jelaskan aliran kerja program.
- Sebutkan semua variable dan parameter yang digunakan beserta tipenya.

6. Buatlah Program yang apabila dijalankan akan menanyakan suhu dalam derajat Celcius, kemudian menampilkan kembali suhu tersebut dalam derajat Fahrenheit, derajat reamur, dan derajat Kelvin. Penggunaan paling sedikit 3 buah prosedur.

Catatan :

C derajat Celcius =  $(9/5 \times C) + 32$  derajat Fahrenheit.

C derajat Celcius =  $(4/5 \times C)$  derajat Reamur

C derajat Celcius =  $(C + 273)$  derajat Kelvin

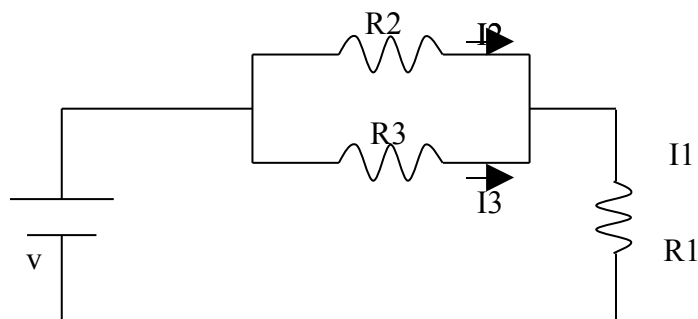
7. Buatlah program untuk menghitung akar-akar suatu persamaan kuadrat  $Ax^2 + Bx + C$ . harga konstanta A, B dan C ditanyakan terlebih dahulu, lalu ketiganya dipakai untuk menghitung diskriminan D.

Jika  $D > 0$  maka terdapat dua akar riil yang berbeda.

Jika  $D = 0$  maka terdapat dua akar riil kembar

Jika  $D < 0$  maka akar-akarnya tidak riil.

8. Perhatikan rangkaian listrik berikut ini



Buat program untuk menanyakan besarnya sumber tegangan V, resistor r1, r2 dan r3 (ohm), lalu menghitung :

- Arus i1, i2, i3 (Ampere)

- b) Daya yang hilang pada masing-masing resistor (Watt).
9. Sebuah benda bermassa  $M$  kg jatuh dari ketinggian  $H$  meter. Buatlah program untuk menghitung :
- a) Waktu yang dibutuhkan untuk mencapai tanah.
  - b) Kecepatan saat peluru mencapai tanah.
10. Tiga buah gaya  $F_1$ ,  $F_2$ , dan  $F_3$  bekerja bersama-sama pada sebuah benda. Buatlah program untuk menanyakan besar dan sudut kerja masing-masing gaya lalu menghitung gaya resultan dari ketiga gaya tersebut.
11. Buatlah program untuk menampilkan dan mengerjakan pilihan menu di layar sebagai berikut :
- Pilihan yang tersedia ---
- A. Menghitung Luas Segitiga
  - B. Menghitung Luas Lingkaran

## **BAB 8 FUNGSI**

### **Pengertian Fungsi**

Seperti halnya prosedur, fungsi adalah modul program, yaitu blok program yang ditulis menurut aturan tertentu, dan digunakan untuk membantu bagian utama program.

Pada bab sebelumnya kalian telah diperkenalkan bahwa pada Pascal memiliki beberapa FUNGSI STANDARD, seperti : Abs, Sqr, Exp, Ln dan lain-lain. Fungsi-fungsi tersebut dikatakan standard karena memang sudah dibuat dan tersimpan di dalam Compiler Turbo Pascal. Pada saat diperlukan, anda tinggal memanggil saja.

Selain menyediakan fungsi-fungsi standard, Turbo Pascal memberi fasilitas kepada anda untuk menciptakan fungsi sendiri, fungsi-fungsi yang anda ciptakan tersebut FUNGSI BUATAN (User Defined Function). Untuk selanjutnya, yang dimaksud dengan fungsi dalam buku ini adalah fungsi buatan, kecuali apabila disebutkan sebelumnya.

Pada dasarnya, cara menciptakan suatu fungsi sama dengan cara membuat prosedur. Struktur keduanya sama. Perbedaan antara prosedur dengan fungsi buatan hanya terletak pada system pertukaran data serta tata cara pemberi namanya saja. Seperti halnya fungsi standard, fungsi buatan dapat menerima beberapa data, tetapi hanya dapat menghasilkan satu keluaran saja.

Fungsi buatan diletakkan sesudah bagian deklarasi (konstanta, type variable) tetapi sebelum blok program utama. Anda dapat mendeklarasikan fungsi pada daerah yang sama dengan deklarasi prosedur.

Fungsi buatan yang sudah diciptakan dapat dipanggil pada saat anda berada di blok program utama, dengan cara menuliskan nama fungsi tersebut, diikuti dengan argument-argumen yang akan dimasukkan.

Pada saat fungsi buatan dipanggil, perjalanan program akan meloncat ke awal fungsi yang dimaksud. Data-data yang ada di dalam parameter

akan diolah sehingga diperoleh sebuah hasil. Hasil tersebut akan dibawa kembali ke tempat dimana fungsi tersebut dipanggil.

### Contoh 8-1

Pada Pascal memiliki fungsi standard yang digunakan untuk menghitung kuadrat suatu bilangan, yaitu fungsi `sqr`, fungsi ini dapat menerima beberapa argument, tetapi hanya menghasilkan sebuah keluaran saja.

Berikut ini akan dibuat fungsi buatan yang memiliki kegunaan yang sama dengan fungsi `sqr`, yaitu menghitung kuadrat dari argument.

Program Contoh\_Fungsi

VAR

    a, b, c : Integer;

    p, q : Real;

FUNCTION Kuadrat (zz : Integer) : real;

BEGIN

    Kuadrat

END;

Fungsi  
kuadrat

BEGIN

    Writeln ('Masukkan tiga bilangan a, b, c : ');

    Readln (a, b, c);

    p := Kuadrat (a);

    writeln (' a kuadrat = ', p);

    writeln (' b kuadrat = ', Kuadrat (b));

    writeln ('b kuadrat = ', Sqr (b));

    q := kuadrat (a + b + c);

    writeln (q);

Bagian  
Utama  
Program

END.

Perhatikan bahwa fungsi kuadrat memiliki struktur yang sama dengan sebuah prosedur. Fungsi buatan yang bernama kuadrat ini memiliki sifat

yang sama dengan fungsi standar `sqr`. Setiap kali fungsi kuadrat dipanggil, argumennya akan disalin ke parameter `zz`, lalu aliran program akan meloncat ke awal blok fungsi kuadrat, dan melakukan proses disana. Setelah selesai, hasil yang diperoleh akan dibawa kembali ke bagian utama program.

### **Deklarasi Fungsi**

Pembuatan suatu fungsi mirip dengan pembuatan prosedur. Berikut ini beberapa hal penting yang harus diperhatikan pada saat menciptakan fungsi buatan sendiri.

(a) Sama seperti prosedur, fungsi harus diberi judul. Judul suatu fungsi diawali oleh kata `Function` dan diikuti nama fungsi yang akan dibuat, serta jenis data yang akan dihasilkan oleh fungsi tersebut.

Nama suatu fungsi harus diawali huruf, tidak mengandung spasi, dan tidak lebih dari 63 karakter. Selain itu, nama suatu fungsi harus sama dengan nama variable yang akan digunakan untuk menyimpan proses.

### **Contoh 8-2**

Menulis nama fungsi,

```
FUNCTION KUADRAT (zz : Integer) : Real;
```

```
FUNCTION VOLUME (p, l, t : Integer) : Integer;
```

Setelah menuliskan nama fungsi, anda harus menyebutkan semua parameter yang akan digunakan oleh fungsi tersebut beserta tipenya (di dalam kurung), fungsi buatan harus memiliki parameter (berbeda dengan prosedur, yang bisa bekerja tanpa memakai parameter). Parameter di dalam fungsi buatan dapat disamakan dengan argument pada fungsi standard. Parameter akan diisi dengan data-data mentah yang akan diolah oleh fungsi yang bersangkutan.

Setelah semua parameter disebutkan, terdapat dua titik yang dilanjutkan dengan jenis data yang akan dihasilkan oleh fungsi tersebut. (Ingat : Fungsi bias menerima beberapa argument, tetapi hanya bisa mengeluarkan sebuah fungsi saja).

Pada contoh 8-1 di atas, fungsi kuadrat memiliki sebuah parameter/ argument, yaitu zz (integer), dan hasil yang dikeluarkan oleh fungsi kuadrat. Adalah data real. Fungsi volume memiliki tiga buah parameter/ argument, yaitu p, l dan t (Integer) dan hasil yang dikeluarkan adalah data integer.

(b) Suatu fungsi dapat memiliki beberapa argument, tetapi hanya dapat mengeluarkan satu hasil saja. Data yang dihasilkan oleh sebuah fungsi harus disimpan didalam variable yang namanya sama dengan anam fungsi tersebut.

Contoh 8-3

```
FUNCTION Kuadrat (zz : Integer) : real;
```

```
BEGIN
```

```
    Kuadrat := zz * zz;
```

```
END;
```

```
FUNCTION Volume (p, l, t : Integer) : Integer;
```

```
BEGIN
```

```
    Volume := p * l * t;
```

```
END;
```

Perhatikan contoh 8-3 di atas. Hasil yang dikeluarkan oleh fungsi kuadrat harus disimpan di dalam variable yang bernama kuadrat pula. Hasil yang dikeluarkan oleh fungsi volume harus disimpan didalam variable yang bernama volume pula.

Fungsi yang akan dipergunakan harus sudah dibuat terlebih dahulu. Cara memanggil fungsi adalah dengan menyebutkan namanya, dan diikuti deretan harga yang akan dimasukkan ke dalam parameter/ argument fungsi tersebut.

Seperti halnya 'prosedur', fungsi juga dapat memiliki bagian deklarasi local sendiri. Aturan ruang lingkup deklarasi global dan deklarasi local pada prosedur juga berlaku pada fungsi.

Hal yang perlu diingat adalah, walaupun suatu fungsi boleh memiliki konstanta, tipe, maupun variable local, namun nama variable yang

digunakan untuk menyimpan hasil akhir harus tetap sama dengan nama fungsi tersebut.

#### Contoh 8-4

Fungsi rata-rata berikut ini digunakan untuk mencari rata-rata tiga buah bilangan, Fungsi ini memiliki tiga buah argument a, b, dan c.

```
FUNCTION Ratarata (a, b, c : Integer) : Real;  
CONST  
    zz = 3;  
VAR  
    Jumlah : Integer;  
    Hasil : Real;  
BEGIN  
    Jumlah := a + b + c;  
    Hasil := jumlah/ zz;  
    Ratarata := Hasil;  
END;
```

Oleh karena nama fungsi adalah Ratarata, maka hasil perhitungan ratarata dari ketiga parameter/ argument a, b dan c harus disimpan di variable rata-rata pula.



### Contoh 8-5

Program menggunakan fungsi yang memiliki deretan local.

PROGRAM Mencari\_Bilangan\_Terbesar;

VAR

a, b, c : Integer;

FUNCTION Maksimum (p, q, r : Integer) : Integer;

VAR

Big : Integer;

Deklarasi  
lokal

Blok  
Fungsi  
maksimum

BEGIN

Big := p;

If q > big then big := q;

If r > big then big := r;

Maksimum := big;

END;

BEGIN

Write ('Tuliskan tiga bilangan a, b, c : ');

Readln (a, b, c);

Terbesar := Maksimum (a, b, c);

Writeln;

Writeln('Bilangan terbesar =', Terbesar);

Bagian  
Utama  
program

END.

Variabel a, b dan c adalah variable global, sedang Big adalah variable local. Variable big hanya dikenali oleh blok fungsi Maksimum.

### Fungsi Berganda

Seperti halnya prosedur, suatu fungsi dapat kalian panggil selagi sedang berada dalam fungsi yang lain, tentu saja, fungsi yang dipanggil tersebut harus sudah, dibuat dahulu sebelumnya.

### Contoh 8-6

Program berikut ini menunjukkan bahwa suatu fungsi dapat dipanggil oleh fungsi yang lain.

```
PROGRAM contoh_fungsi_berganda;
```

```
VAR
```

```
    a, b, c : Integer;
```

```
    FUNCTION Jumlah (x, y, z : Integer) : Integer;
```

```
    BEGIN
```

```
        Jumlah := x + y + z;
```

```
    END;
```

Fungsi  
jumlah

```
    FUNCTION Ratarata (x, y, z : Integer) : Integer;
```

```
    VAR
```

```
        Sum : Integer;
```

```
    BEGIN
```

```
        Sum := jumlah (x, y, z);
```

```
        Ratarata := sum/3;
```

```
    END;
```

Fungsi  
Ratarata

```
BEGIN
```

```
    Write ('Tuliskan tiga bilangan a, b, c : ');
```

```
    Readln (a, b, c);
```

```
    Writeln ('Rata-ratanya = ', ratarata (a,b,c)) ;
```

```
END.
```

Bagian  
Utama  
program

Program di atas akan membaca deretan tiga buah data a, b dan c dari keyboard. Fungsi rata-rata dipanggil dari bagian utama program untuk menghitung rata-rata dari tiga bilangan a, b dan c.

Fungsi rata-rata sendiri pada saat bekerja memanggil fungsi jumlah untuk menghitung jumlah tiga data yang tersimpan di dalam parameter x, y dan z.

Suatu fungsi dapat dideklarasikan di dalam fungsi lain. Fungsi yang dideklarasikan secara local hanya dikenali oleh blok fungsi tempat deklarasi dilakukan.

Contoh 8-7 :

Fungsi jumlah pada contoh 8.6 dapat diletakkan di dalam fungsi ratarata.

```
PROGRAM contoh_fungsi_berganda;
VAR
    a, b, c : Integer;
    FUNCTION ratarata (u, v, w : Integer) : real;
    VAR
        Sum : Integer;
        FUNCTION jumlah (x, y, z : Integer) : Integer;
        BEGIN
            Jumlah := x + y + z;
        END;
    BEGIN
        Sum := jumlah (u, v, w);
        Ratarata := sum/3;
    END;
BEGIN
    Write ('tuliskan tiga bilangan a, b, c : ');
    Readln (a, b, c);
    Writeln ('Rata-ratanya = ', Ratarata(a, b, c));
END.
```

Blok  
Ratarata

Fungsi  
Jumlah

Bagian  
Utama  
Program

Perhatikan bahwa fungsi jumlah terletak di dalam fungsi ratarata. Oleh karena itu, fungsi jumlah disebut memiliki ruang lingkup local terhadap fungsi ratarata. Fungsi jumlah hanya dapat dipanggil dari dalam blok

fungsi rata-rata. Anda dapat memanggil fungsi jumlah dari bagian utama program.

Fungsi jumlah hanya akan dijalankan apabila fungsi rata-rata dijalankan. Kesimpulan yang dapat diperoleh adalah bahwa suatu fungsi hanya dapat dipanggil dari dalam blok tempat fungsi tersebut berada.

Variable a, b, c adalah variable global, sedang variable Sum adalah variable local yang hanya dikenali oleh blok fungsi rata-rata. Variable sum dikenali oleh blok fungsi, jumlah sebab fungsi jumlah berada di dalam blok fungsi rata-rata.

### **Gabungan Prosedur dan Fungsi**

Fungsi dan prosedur dapat digunakan bersama-sama di dalam sebuah program, fungsi dapat diletakkan di dalam suatu prosedur, demikian pula sebaliknya, prosedur dapat diletakkan di dalam suatu fungsi. Selain itu fungsi dan prosedur dapat pula diletakkan terpisah satu sama lain.

Pada saat menggunakan fungsi dan prosedur bersama-sama, anda harus selalu mengingat kembali hal sebagai berikut :

- Ruang lingkup variable global dan variable local yang anda pergunakan.
- Proses yang dilakukan oleh suatu prosedur dapat menghasilkan lebih dari satu keluaran, sedang proses yang dilakukan oleh fungsi hanya akan menghasilkan satu keluaran saja.
- Nama suatu prosedur tidak boleh sama dengan variable yang digunakan, tetapi nama suatu fungsi justru harus sama dengan nama variable yang digunakan untuk menyimpan hasil akhir proses.

Perhatikan beberapa contoh pemakaian prosedur bersama-sama dengan fungsi berikut ini.

#### **Contoh 8-8**

Program berikut menunjukkan bahwa suatu fungsi dapat dipanggil oleh suatu prosedur.

PROGRAM Menentukan\_Lulus\_Tidak\_Lulus;

Var

a, b : Integer;

```
FUNCTION Ratarata (x, y : Integer) : Real;  
BEGIN  
    Ratarata := (x + y)/2;  
END;
```

Fungsi  
Ratarata

```
FUNCTION Lulus (zz : Integer) : Boolean;  
BEGIN  
    IF zz > 55 then lulus := True;  
    ELSE Lulus := False;  
END;
```

Fungsi  
Kelulusan

<pre> PROCEDURE Kelulusan (x, y : Integer); VAR     NilaiAkhir : Real;     OK : Boolean; BEGIN     NilaiAkhir := Ratarata (x, y);     OK := Lulus (Nilai Akhir);     IF OK = True THEN writeln ('Lulus');     ELSE Writeln ('Tidak Lulus'); END; </pre>	<div style="border-left: 1px solid black; height: 150px; margin-left: 5px;"></div> Prosedur Kelulusan
<pre> BEGIN     Writeln ('Tuliskan Nilai1, Nilai2 : ');     Readln (a, b);     Kelulusan (a, b); END. </pre>	<div style="border-left: 1px solid black; height: 80px; margin-left: 5px;"></div> Bagian Program utama

Fungsi rata-rata dan fungsi lulus dipanggil oleh prosedur kelulusan. Agar fungsi rata-rata dan fungsi lulus dapat dipanggil dari prosedur kelulusan, maka keduanya harus diletakkan sebelum prosedur kelulusan.

Variable a dan b adalah variable global, sedang nilaiakhir dan Ok adalah variable local yang hanya dikenali oleh blok prosedur kelulusan.

Contoh 8-9

Program berikut menunjukkan bahwa suatu fungsi dapat diletakkan di dalam prosedur.

PROGRAM menghitung\_pajak;

Var

Gaji : Real;

PROCEDURE HitungPajak (a : real);

VAR

Persen, pajak : real;

```

FUNCTION Persentase (zz : real) : real;
BEGIN
    IF zz < 100000 THEN Persentase := 0.01
    ELSE Persentase := 0.03;
END;

BEGIN
    Persen := Persentase (a);
    Pajak := a * persen;
    Writeln ('Besarnya gaji := Rp. ', a);
    Writeln ('Besarnya Pajak := Rp. ', pajak);
END;
BEGIN
    Write ('Tuliskan gaji anda : ');
    Readln (Gaji);
    HitungPajak (gaji);
END.

```

Fungsi persentase terletak di dalam prosedur HitungPajak, sehingga fungsi persentase hanya dikenali oleh blok prosedur HitungPajak saja, Fungsi persentase tidak dapat dipanggil oleh bagian utama program.

Gaji adalah variable global, sedang persen dan pajak adalah variable local yang hanya dikenali oleh blok prosedur hitungpajak. Fungsi persentase dapat mengenali variable persen dan pajak karena fungsi persentase berada di dalam blok prosedur hitungpajak.

#### Contoh 8-10

Program berikut menunjukkan bahwa suatu prosedur dapat diletakkan di dalam fungsi.

PROGRAM Menghitung\_pajak;

VAR

Gaji : real;

FUNCTION Pajak (a : real) : real;

VAR

Persen, Pajak : Real;

PROCEDURE MenghitungPersen;

BEGIN

IF a < 100000 THEN Persen := 0.01

ELSE Persen := 0.03;

END;

BEGIN

MenghitungPersen (a);

Pajak := a \* persen;

END;

BEGIN

Write ('Tuliskan gaji anda : ');

Readln(gaji);

Writeln ('Besarnya Gaji := Rp. ', a);

Writeln ('Besarnya Pajak := Rp. ', Pajak(gaji));

END.

Prosedur menghitungpersen terletak di dalam fungsi pajak, sehingga prosedur menghitungpersen hanya dikenali oleh blok fungsi pajak saja. Bagian utama program tidak dapat memanggil prosedur menghitung persen.



Gaji adalah variable global, sedang persen dan pajak adalah variable local yang hanya dikenali oleh blok fungsi pajak saja. Kedua variable local tersebut dapat dikenali oleh prosedur menghitungpersen sebab prosedur menghitungpersen berada di dalam blok fungsi pajak.

### Soal

1. Perhatikan beberapa judul fungsi di bawah ini, Tentukan manakah judul fungsi yang benar !
  - a) FUNCTION Maksi;
  - b) FUNCTION Isi (A: B : Integer) : Integer;
  - c) FUNCTION Isi (A : Boolean) : Char;
  - d) FUNCTION Maksimum(A, B : Real :C) : Integer;
  - e) FUNCTION ab cd (A:Integer) : Char;
  - f) FUNCTION zkali Y (Z real , Y Real) : Char;
  - g) FUNCTION Pendapatan(Gaji : Real; Kode : String[B]) : String[20];
2. Tuliskan judul fungsi untuk keperluan berikut ini :
  - a) Fungsi riil bernama AbagiB dengan dua buah parameter riil a dan b.
  - b) Fungsi integer bernama Terkecil dengan tiga buah parameter integer A, B dan C.
  - c) Fungsi Riil bernama Pajak dengan parameter KodeWilayah (Integer), Gaji (riil), dan status (Character).
3. Buatlah sebuah fungsi untuk mencari bilangan terbesar dari tiga bilangan integer a, b dan c.
4. Buatlah sebuah fungsi Boolean dengan sebuah parameter integer. Fungsi tersebut akan menghasilkan nilai TRUE apabila parameternya bilangan GANJIL.
5. Buatlah sebuah fungsi untuk menghitung luas permukaan bola dengan rumus :  $L = 4 \pi r^2$
6. Buatlah sebuah fungsi riil untuk menghitung rata-rata tiga buah bilangan integer x, y dan z.

7. Buatlah program untuk menanyakan sisi-sisi tegak sebuah segitiga siku-siku, lalu menghitung sisi miringnya dengan menggunakan sebuah fungsi.

8. Sin adalah fungsi standar yang sudah tersedia untuk menghitung sinus suatu sudut, argument fungsi ini harus dalam satuan radian. Buatlah program untuk menanyakan sudut dalam derajat lalu menghitung sinus sudut tersebut

Petunjuk :

Buatlah sebuah fungsi buatan untuk mengubah satuan sudut dari derajat ke radian, lalu menghitung sinus sudut tersebut.

9. Buatlah program untuk mencari huruf seorang murid. Nilai angka ditanyakan terlebih dahulu. Nilai angka yang dimasukan harus diantara 0 s/d 100. pemasukkan nilai angka akan diulang apabila nilai yang dimasukkan tidak memenuhi syarat.

10. Buatlah program untuk membaca sederetan data angka bulat yang dimasukkan dari keyboard terus menerus. Pembacaan data angka ini akan berakhir apabila yang diketikkan adalah angka 0 (nol). Program tersebut menghitung banyaknya bilangan ganjil serta bilangan genap yang dimasukkan.

11. Buatlah program untuk menanyakan pendapatan seseorang lalu menghitung besarnya pajak pendapatan yang harus dibayar.

- apabila pendapatannya kurang dari Rp. 500000,- maka prosentase pajak yang harus dibayar adalah 2.5 persen.
- Apabila pendapatannya diantara Rp. 500000,- sampai dengan Rp. 1000000,- maka prosentase pajak yang harus dibayar 3 persen.
- Apabila pendapatannya lebih dari Rp. 1000000, maka prosentase pajak yang harus dibayar 3.5 persen.

12. Besarnya persentase pajak pendapatan untuk DKI Jakarta adalah 5 persen, untuk Jawa Barat 3.5 persen. Untuk Jawa Tengah 3 persen, dan untuk Jawa Timur 2.75 persen. Buatlah program untuk

menanyakan nama, pendapatan, serta wilayah tempat tinggal seseorang, lalu menghitung pajak yang harus dibayar.

## **LATIHAN ULANGAN BAB 6, 7, 8**

### **PILIHAN GANDA**

1. Manakah yang salah dari deklarasi di bawah ini :
  - a. Function abc(a:byte):real;
  - b. Function xyz(a:byte):byte;
  - c. Function klm(k,l:byte):real;
  - d. Function stu(s:real);
  - e. Salah semua
2. Manakah yang salah dari deklarasi di bawah ini :
  - a. Procedure abc;
  - b. Procedure abc(a:byte);
  - c. Procedure xyz(var x:integer);
  - d. Procedure klm(a,b:real);
  - e. Salah semua

3. Penggalan deklarasi program sebagai berikut :

Type ar = array [1..10] of byte;

Var a:ar; l, b:byte;

Procedure Baca(var x:ar);

Begin

    For i:=1 to 10 do readln(x[i]);

End;

Function jumlah(x:ar):byte;

Var j:byte;

Begin

    j:=0;

    For i:=1 to 10 do

        j:=j+x[i] mod l;

End;

Begin

    Baca(a); b:=Jumlah(a);

    Writeln(b);

End.

Bila data yang dibaca berurutan sebagai berikut : 1, 2, 3, 5, 6, 7, 8, 9, 10 maka hasil output yang dicetak :

- a. 10
- b. 55
- c. 0
- d. 5
- e. Salah semua

4. Masih berhubungan dengan no. 3 di atas. Bila data yang dibaca secara berurutan : 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, maka output yang tercetak :

- a. 10
- b. 22
- c. 20
- d. 30
- e. Salah semua

5. Masih berhubungan dengan nomor 3 di atas. Bila judul procedure Baca (var x:ar) diubah menjadi procedure Baca(x:ar) dan menggunakan data pada soal no. 14, maka output yang tercetak :

- a. 10
- b. 22
- c. 0
- d. 30
- e. Salah semua

6. Penggalan program Pascal sebagai berikut :

```
Function Fakto(n:byte):longint;
```

```
Begin
```

```
  If n=0 then
```

```
    Fakto:=0
```

```
  Else
```

```
    Fakto:=n*Fakto(n-1);
```

```
End;
```

Bila function di atas dipanggil dengan statemen  $y:=\text{fakto}(5)$ ; maka  $y$  bernilai :

- a. 0
- b. 5
- c. 120
- d. 60
- e. Salah semua

7. Masih berhubungan dengan no. 6, di atas, bila statemen  $\text{fakto}:=n*\text{fakto}(n-1)$  diganti dengan  $\text{fakto}:=n+\text{fakto}(n-1)$  dan dipanggil dengan  $y:=\text{fakto}(6)$  maka  $y$  berharga

- a. 21
- b. 24
- c. 6
- d. 20
- e. Salah semua

8. Berikut ini penggalan program Pascal :

```
Function Abc(a,b:byte):longint;
```

```
Begin
```

```
  If b=0 then Abc:=1
```

```
  Else Abc:=a*Abc(a, b-1);
```

End;

Bila function abc di atas dipanggil dengan  $x:=\text{Abc}(5,3)$ ; maka x berharga

- a. 15
- b. 125
- c. 1
- d. 0
- e. Salah semua

9. Masih berhubungan dengan nomor 8, function abc dipanggil dengan  $x:=\text{abc}(3,5)$  maka berharga :

- a. 81
- b. 15
- c. 243
- d. 25
- e. Salah semua

10. Masih berhubungan dengan soal nomor 8, bila statemen  $\text{abc}:=a*\text{abc}(a, b-1)$ ; diganti dengan  $\text{abc}:=a+\text{abc}(a, b-1)$ ; dan dipanggil dengan  $x:=\text{abc}(3,4)$ ; maka berharga :

- a. 13
- b. 12
- c. 10
- d. 5
- e. Salah semua

11. Penggalan program pascal sebagai berikut :

```
Type kal=string[20];  
Procedure pgr(x:kal,n:byte);  
Begin  
    If n>0 then
```

```

Begin
Write (x[n]);
Pgr(x, n-1);
End;

```

End;

Bila dipanggil procedure dengan pgr("tTOKI97",6), maka outputnya :

- a. TOKI97
- b. 79IKOT
- c. TOKI
- d. TOK97I
- e. Salah semua

12. Bila kata:string[20]='Kasur nababan rusak'; procedure pgr no. 26 di atas di panggil dengan statemen pgr(kata, length(kata)); maka outputnya :

- a. Rusak kasur nababan
- b. Nababan kasur rusak
- c. Kasur nababan rusak
- d. Rusak nababan kasur
- e. Salah semua

13. Penggalan program sebagai berikut :

```
Function Xyz(n:byte):integer;
```

```
Begin
```

```
  If n=1 then Xyz:=n
```

```
  Else Xyz:=1+Xyz(n-1);
```

```
End;
```

Bila function Xyz dipanggil dengan a:=Xyz(10); maka a berharga :

- a. 1
- b. 9
- c. 10
- d. 11



e. Salah semua

14. Masih berhubungan dengan nomor 13, bila dipanggil dengan `y:=Xyz(16)`; maka `y` berharga :

a. 15

b. 5

c. 1

d. 0

e. Salah semua

15. Penggalan program sebagai berikut :

```
Procedure abc(A,B,C,D,E,F:byte);
```

```
Begin
```

```
  C:=A + B;
```

```
  D:=A - B;
```

```
  E:=A*B;
```

```
  F:A div B;
```

```
End;
```

Bila procedure `abc` dipanggil dengan `abc(10,3,x,y,z,w)` maka `x,y,z` dan `w` masing-masing berharga:

a. 13, 7, 30, 3

b. 0, 0, 0, 0

c. 13, 7, 30, 3, 33

d. 1, 1, 1, 1

e. Salah semua

16. Penggalan program :

```
Procedure tulis(kata:string[80]);
```

```
Var i:byte;
```

```
Begin
```

```
For i:=length downto 1 do write(kata[i]);  
End;
```

Bila procedure tulis dipanggil dengan tulis('Tim Olimpiade Komputer Indonesia'); maka hasilnya :

- a. Tim Olimpiade Komputer Indonesia
- b. Indonesia Komputer Olimpiade Tim
- c. Aisenodni retupmoK edalipmilO miT
- d. Tim Inoonesia Olimpiade Komputer
- e. Salah semua

17. Masih berhubungan dengan no. 45, bila for i:=length(kata) downto 1 do diganti dengan for i:=1 to length(kata) do maka hasilnya :

- a. Tim Olimpiade Komputer Indonesia
- b. Indonesia Komputer Olimpiade Tim
- c. Aisenodni retupmoK edalipmilO miT
- d. Tim Inoonesia Olimpiade Komputer
- e. Salah semua

**Penggalan program sebagai berikut :**

```
Type ar=array[1..3,1..3] of integer;  
Var a,b,c:ar; i,j,k:byte;  
.....  
procedure abc(var x:ar);  
begin  
    for i:=1 to 3 do  
        for j:=1 to 3 do read(x[i,j]);  
    end;  
procedure pqr(x, y:ar; var z:ar);  
begin  
    for i:= 1 to 3 do
```

```

        for j:=1 to 3 do begin z[i,j]:=0;
            for k:=1 to 3 do
                z(i,j):=z(l,j)+x[i,k]*y[k,j];
            end;
        end;
end;
procedure klm(x:ar);
begin for i:=1 to 3 do
    for j:=1 to 3 do
        write(x[i,j]:5;
    writeln;
end;

```

18. Bila main program procedure dipanggil dengan abc(a); dan data yang dimasukkan : 1 2 3 4 5 6 7 8 9, maka matrix a berharga :

a. 1 2 3

4 5 6

7 8 9

b. 1 4 7

2 5 8

3 6 9

c. 1 3 5

2 4 6

7 8 9

d. 1 2 3

7 6 9

4 5 8

e. Salah semua

19. masih berhubungan dengan penggalan program di atas, bila statemen secara berurutan dalam main program sebagai berikut : abc(a); abc(b); pqr(a,b,c); klm(c); maka program ini menyelesaikan :

a. membaca matrix a dan b, mengalikan matrix a dan b menghasilkan matrix c, dan mencetak matrix c

- b. Membaca matrix a dan b, menambahkan matrix a dan b menghasilkan matrix c dan mencetak matrix c
- c. Membaca matrix a dan b, mencetak matrix c
- d. Mencetak matrix c
- e. Salah semua

20. Masih berhubungan dengan nomor 18 dan 19.

Bila data yang dibaca secara berurutan adalah 1 2 1 3 4 1 2 2 3 1 0 0 0 1 0 0 0 1, maka matrikx C berharga :

- a. 3 4 1  
2 2 3  
1 2 1
- b. 1 2 1  
3 4 1  
2 2 3
- c. 2 2 3  
3 4 1  
1 2 1
- d. 3 4 1  
1 2 1  
2 2 1
- e. Salah semua

