**MODUL IV**

**STRUKTUR KENDALI**

**PERCABANGAN BERSYARAT DENGAN**

**IF ... THEN DAN IF ... THEN ... ELSE ...**

* + 1. **TUJUAN**

1. Memahami salah satu jenis statement kendali untuk pemilihan proses
2. Memahami proses pengambilan keputusan dengan statement if ... then ... else ...
3. Memahami variasi proses pengambilan keputusan dalam bentuk tunggal
4. Mampu mengaplikasikan proses pengambilan keputusan dalam program
   * 1. **DASAR TEORI**

Sebuah program sering kali harus dapat memilih pernyataan mana yang akan dikerjakan berdasarkan suatu kondisi logika tertentu, misalnya bagaimana membuat program untuk mengetahui bilangan mana yang lebih besar di antara 2 (dua) buah bilangan.

Untuk menyelesaikan persoalan di atas, Pascal menyediakan fasilitas pengambilan keputusan, yaitu dengan statement IF ... THEN ... ELSE ...

Bentuk umum untuk IF ... THEN ...

IF (ekspresi logika) THEN

BEGIN

bagian yang dikerjakan

jika ekspresi logika bernilai benar (TRUE)

END;

Bentuk umum untuk IF ... THEN ... ELSE ...

IF (ekspresi logika) THEN

BEGIN

bagian yang dikerjakan

jika ekspresi logika bernilai benar (TRUE)

END {tanpa titik koma}

ELSE

BEGIN

bagian yang dikerjakan

jika ekspresi logika bernilai salah (FALSE)

END;

* + 1. **LATIHAN SOAL**

1. **Nomor Program 0401**
   1. Tulislah program berikut ini, kemudian amati outputnya!

**PROGRAM tanya;**

**USES CRT;**

**VAR a : real;**

**BEGIN**

**CLRSCR;**

**WRITE(‘Masukkan sebuah bilangan di bawah 100! ’);**

**READLN(a);**

**IF a<100 THEN**

**WRITE(‘Terima Kasih’);**

**READLN;**

**END.**

* 1. Apa yang terjadi jika anda memasukkan nilai kurang dari 100, jelaskan!

Maka ia akan menampilkan tulisan “Terima Kasih”.

* 1. Apa yang terjadi jika anda memasukkan nilai lebih dari 100, jelaskan!

Ia tidak akan menampilkan output apapun.

1. **Nomor Program 0402**
   1. Tulislah program berikut ini, kemudian amati outputnya!

**PROGRAM nilai\_maksimal;**

**USES CRT;**

**VAR a, b, max : integer;**

**BEGIN**

**CLRSCR;**

**TEXTCOLOR(lightblue);**

**WRITE(‘Masukkan bilangan pertama : ’);READLN(a);**

**WRITE(‘Masukkan bilangan kedua : ‘);READLN(b);**

**IF a>b THEN**

**max:=a**

**ELSE**

**max:=b;**

**WRITELN(‘Bilangan terbesar adalah ‘,max);**

**READLN;**

**END.**

* 1. Modifikasilah program di atas sehingga output yang ditampilkan adalah nilai terbesar adalah bilangan pertama (jika bilangan pertama memiliki nilai terbesar) atau nilai terbesar adalah bilangan kedua (jika bilangan kedua memiliki nilai terbesar)!

..................................................................................................................

..................................................................................................................

..................................................................................................................

..................................................................................................................

..................................................................................................................

..................................................................................................................

..................................................................................................................

..................................................................................................................

* 1. Gantilah bagian-bagian tertentu dari program di atas sehingga dapat digunakan untuk menampilkan bilangan yang memiliki nilai terkecil!

..................................................................................................................

..................................................................................................................

..................................................................................................................

..................................................................................................................

..................................................................................................................

..................................................................................................................

..................................................................................................................

..................................................................................................................

1. **Nomor Program 0403**
   1. Tulislah program berikut ini, kemudian amati outputnya!

**PROGRAM nilai\_ujian;**

**USES CRT;**

**VAR nilai : real;**

**nil\_huruf : char;**

**BEGIN**

**CLRSCR;**

**WRITE(‘Tulis nilai ujian anda (0-100) : ‘);**

**READLN(nilai);**

**SOUND(1000); DELAY(1000); NOSOUND;**

**IF nilai>=80 THEN**

**nil\_huruf:=’A’**

**ELSE IF nilai>=70 THEN**

**nil\_huruf:=’B’**

**ELSE IF nilai>=60 THEN**

**nil\_huruf:=’C’**

**ELSE IF nilai>=20 THEN**

**nil\_huruf:=’D’**

**ELSE**

**nil\_huruf:=’E’;**

**WRITELN(‘Nilai Anda : ‘,nil\_huruf);**

**READLN;**

**END.**

* 1. Modifikasilah program di atas sehingga apabila user memasukkan angka di atas 100, maka program akan memberi pesan kesalahan!

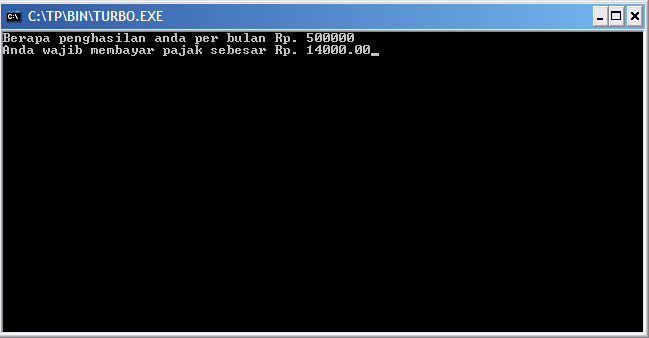
1. **Nomor Program 0404**

Buatlah program yang akan menampilkan pesan apakah bilangan (0-65535) yang user masukkan adalah bilangan genap atau ganjil!

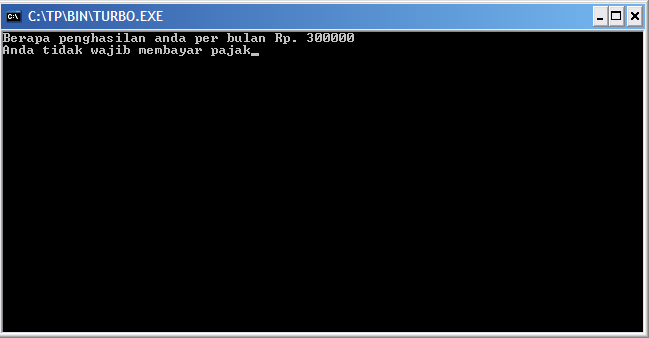
1. **Nomor Program 0405**

Seorang karyawan bekerja di perusahaan X. Karyawan tersebut digaji dengan mengikuti ketentuan PTKP (Penghasilan Tidak Kena Pajak) dan PKP (Penghasilan Kena Pajak). PTKP berlaku bagi karyawan yang digaji di bawah 360000 per bulan, sedangkan bagi karyawan yang gajinya adalah 360000 ke atas akan tergolong PKP, dengan pajak 10% dari Penghasilan Kena Pajak (PKP). PKP = (Gaji-360000). Misalnya seorang karyawan digaji sebesar 500000, maka PKP = 500000 – 360000 = 140000. Jadi, pajak yang harus dibayar = 140000 x 10% = 14000.

Berdasarkan kasus di atas, buatlah program untuk menentukan pajak yang harus dibayar oleh karyawan, yang mana ketika program dijalankan, program akan meminta masukan dari user berupa penghasilan perbulan. Berikut ini contoh output program :



dan



Pengertian percabangan if else

Percabangan if else digunakan untuk membuat

percabangan dimana jika pernyataan benar maka

akan lanjut ke stetmen pertama jika salah maka

akan lanjut ke pernyataan selanjutnya.

Bentuk umum

If – kondisi – statment

Else

If – kondisi2 - statment

Pengertian percabangan case of

Percabangan case of adalah percabanga yang

berfungsi untuk melakukan seleksi atas beberapa

pilihan dengan kondisi yang harus terpenuhi

bentuk umum

case – variabel of - kondisi - statment

Pengertian perulangan for to do

Perulangan for to do adalah perulagan yang akan

mengulang sampai mencapai batas yang ditertentukan

contoh for i:= 1 to 10 do. Artinya prongram akan

mengulang sampai kondisi atau batas terpenuhi.

Pengertian perulangan repeat until

Perulangan repeat until adalah perulangan yang akan

mengulang sampai program bernilai salah contoh

repeat until i<10 maka program akan terus mengulang

jika i masih lebih kecil dari 10.

Pengertian perulangan while do

Perulangan while do adalah perulangan yang akan

terus melakukan perulangan selama kondisi bernilai

benar

Bentuk umum while – variabel - kondisi