

# Job Sheet

## Untuk SMK Jurusan RPL

### Algoritma Dasar & Pemrograman

Oleh :

Isa Hamdan

[isa@my-itb.com](mailto:isa@my-itb.com)

0819 3848 3780

## 1. Output Program

Untuk menampilkan output atau keluaran program pada Pascal menggunakan perintah :

**Write** : menampilkan output kemudian kursor masih pada baris yang sama.

**WriteLn** : menampilkan output kemudian kursor pindah baris.

### **Tugas:**

Amati dan catat menggunakan bahasa anda sendiri tampilan program setelah dijalankan. Perhatikan perbedaan fungsi **write** dan **writeln** kemudian simpan pada Ms Word dan dikumpulkan.

Simpan program menggunakan nama contoh1, contoh2, contoh3, dan seterusnya.

#### **Contoh 1 :**

```
begin
    Write('hello');
    Write(' saya belajar pascal ');

    ReadLn();

end.
```

#### **Contoh 2 :**

```
begin
    WriteLn('hello');
    Write('saya belajar pascal ');
    ReadLn();

end.
```

#### **Contoh 3 :**

```
begin
    WriteLn('PROGRAM PERTAMA SAYA');
    Write('Saya anak SMKN 2 Buduran');
    Write(' sedang belajar pascal, ');
    Write(' ternyata pascal');
    Write(' itu mudah');
    Write(' dan menyenangkan');

    ReadLn();

end.
```

#### **Contoh 4:**

```
begin
    WriteLn('PROGRAM PERTAMA SAYA');
    Writeln('Saya anak SMKN 2 Buduran');
    Writeln('sedang belajar pascal, ');
    WriteLn();
    WriteLn();
    Writeln('ternyata pascal');
    Writeln('itu mudah');
    Writeln('dan menyenangkan');

    ReadLn();

end.
```

## 2. Operasi Matematika

Dalam operasi matematika pascal menggunakan beberapa operator yang dapat digunakan untuk proses perhitungannya. Gunakan operator yang ada dalam tabel untuk operasi perhitungan pada program yang dibuat.

Operator	Operasi	Tipe Operand	Tipe Hasil
<b>*</b>	Perkalian	real, real integer, integer real, integer	real integer real
<b>DIV</b>	Hasil Pembagian yang utuh	integer, integer	Integer
<b>/</b>	Pembagian real	real, real integer, integer real, integer	real real real
<b>MOD</b>	Sisa pembagian	integer, integer	Integer
<b>+</b>	Penambahan	real, real integer, integer real, integer	real integer real
<b>-</b>	Pengurangan	real, real integer, integer real, integer	real integer real

### Tugas:

Amati dan catat menggunakan bahasa anda sendiri tampilan program setelah dijalankan. Perhatikan fungsi **matematika** dan **tanda koma (,)** kemudian simpan pada Ms Word dan dikumpulkan.

### Contoh 1:

```
begin
    WriteLn( 3 * 2);
    Writeln( 4 / 3);
    Writeln( 4 / 3 :0:2);
    Writeln(4 div 3);
    Writeln( 4 mod 3);
    Writeln( 4 + 3);
    Writeln( 4 - 3);

    ReadLn();

end.
```

### Contoh 2:

```
begin
    WriteLn( '3 * 2 = ', 3 * 2);
    Writeln( '4 / 3 = ', 4 / 3);
    Writeln( '4 / 3 = ', 4 / 3 :0:2);
    Writeln('4 div 3 = ',4 div 3);
    Writeln('4 mod 3 = ', 4 mod 3);
    Writeln( '4 + 3 = ',4 + 3);
    Writeln('4 - 3 = ', 4 - 3);

    ReadLn();

end.
```

### 3. Input, Variabel, dan Type data

Variabel adalah nama tempat yang digunakan untuk menyimpan data pada memori komputer. Nilai dari suatu variabel dapat berubah sewaktu – waktu ketika program sedang dijalankan.

Jika kita menggunakan pengenalan variabel di dalam program, maka variabel tersebut harus kita deklarasikan terlebih dahulu. Pada waktu kita mendeklarasikan suatu variabel, maka kita harus menentukan tipe dari datanya. Tipe dari data ini menunjukkan suatu nilai yang dapat digunakan oleh variabel bersangkutan. Pada pokok bahasan ini akan dijelaskan bentuk tipe data standar, yaitu dapat berbentuk tipe-tipe data numerik integer, numerik real, karakter, string dan boolean.

#### Tipe data integer

Data numerik integer merupakan nilai bilangan bulat baik dalam bentuk desimal maupun hexadesimal. Nilai integer hexadesimal diawali dengan tanda dollar (\$). Pascal menyediakan 5 macam tipe data integer yang masing-masing mempunyai jangkauan nilai yang berbeda seperti pada tabel berikut :

<b>Type</b>	<b>Ukuran Memori</b>	<b>Jangkauan Nilai</b>
Byte	1 byte	0..255
ShortInt	1 byte	-128..127
Integer	2 byte	-32768..32767
Word	2 byte	0..65535
LongInt	4 byte	-2147483648..2147483647

#### Tipe data Real

Nilai konstanta numerik real berkisar dari 1E-38 sampai dengan 1E+38 dengan signifikansi sampai dengan 11 digit . E menunjukkan nilai 10 pangkat. Nilai konstanta numerik real menempati memori sebesar 6 byte. Berikut tabel tipe data real.

<b>Type</b>	<b>Jangkauan</b>	<b>Ketelitian</b>	<b>Ukuran</b>
Real	2.9e-39..1.7e38	11-12 digit	6 bit
Single	1.5e-45..3.4e38 <sup>3</sup>	7-8 digit	4 bit
Double	5.0e-324..1.7e308	15-16 digit	8 bit
Extended	3.4e-4932..1.1e4932	19-20 digit	10 bit
Comp	-9.2e18..9.2e18	19-20 digit	8 bit

#### Tipe data Karakter (Char)

Nilai data karakter berupa sebuah karakter yang ditulis diantara **tanda petik tunggal**, seperti misalnya 'A', 'a', '!', '%', '5' dan sebagainya. Penggunaan variabel untuk menyimpan data tipe karakter ini harus dideklarasikan dengan **tipe char**.

#### Tipe data String

Nilai data string merupakan urutan dari karakter yang terletak di antara tanda petik tunggal. Bila karakter petik merupakan bagian dari konstanta string, maka dapat ditulis dengan menggunakan dua buah petik tunggal berurutan. Nilai data string akan menempati memori sebesar banyaknya karakter string ditambah dengan 1 byte. Bila panjang dari suatu string di dalam deklarasi variabel tidak disebutkan, maka dianggap panjangnya adalah 255 karakter.

**Tugas:**

Amati dan catat menggunakan bahasa anda sendiri tampilan program setelah dijalankan. Perhatikan fungsi **variabel** dan **posisi variabel** pada output program kemudian simpan pada Ms Word dan dikumpulkan

**Contoh 1 :**

Petunjuk → setelah program dijalankan ketik ' Saya sedang belajar pascal ' kemudian tekan enter.

**Input data**

**Read** : Membaca input data dari keyboard dan kursor masih pada baris yang sama.

**ReadLn** : Membaca input data dari keyboard dan kursor berpindah baris.

```
var
tulisan : string [100];

begin
    Write('ketik tulisan anda : ');
    ReadLn(tulisan);

    WriteLn(tulisan);
    WriteLn(tulisan);
    WriteLn(tulisan);
    WriteLn(tulisan);
    WriteLn(tulisan);

    ReadLn();

end.
```

**Contoh 2 :**

```
var
angka : integer;

begin
    Write('ketik angka anda : ');
    ReadLn(angka);

    WriteLn(angka);
    WriteLn('angka yang anda masukan adalah : ', angka);
    WriteLn(angka, ' adalah angka yang anda masukan ');
    WriteLn('adalah ', angka, ' yang anda masukan ');
    WriteLn(angka);

    ReadLn();

end.
```

## 4. Algoritma Sekuensial

Algoritma sequensial adalah algoritma yang dijalankan secara berurutan dari baris pertama hingga baris terakhir tanpa terjadi pengulangan pada baris atau berpindah ke baris lain yang tidak urut.

### Contoh 1:

Program menghitung luas persegi panjang.

#### Tugas:

Amati dan catat menggunakan bahasa anda sendiri tampilan program setelah dijalankan. Perhatikan ***cara pembacaan variabel*** pada output program kemudian simpan pada Ms Word dan dikumpulkan

### Algoritma

Rumus luas persegi panjang adalah  $\text{Luas} = p * l$ , pada rumus ini terdapat 3 variabel yang digunakan untuk menyimpan data yaitu (Luas, p, l) type data bisa menggunakan integer atau real.

### Program

```
var
Luas, p, l : integer;

begin
    Write('input p : ');
    ReadLn(p);

    Write('input l : ');
    ReadLn(l);

    Luas := p * l ;
    WriteLn('Luas persegi panjang adalah : ', Luas);

    ReadLn();

end.
```

### Contoh 2:

Program menghitung luas Lingkaran.

#### Tugas:

Amati dan catat menggunakan bahasa anda sendiri tampilan program setelah dijalankan. Perhatikan ***cara pembacaan variabel*** pada output program kemudian simpan pada Ms Word dan dikumpulkan

### Algoritma

Rumus luas lingkaran  $\text{Luas} = \text{phi} * r * r$ , pada rumus ini terdapat 2 variabel dan 1 konstanta yang digunakan untuk menyimpan data yaitu (Luas, r) type data yang digunakan **real**.

### Program

```
const
phi = 3.14;

var
Luas, r : real;

begin
    Write('input r : ');
    ReadLn(r);

    Luas := phi * r * r ;
    WriteLn('Luas lingkaran adalah : ', Luas:0:2);

    ReadLn();

end.
```

### Contoh 3:

Program menghitung konversi suhu.

### Tugas:

Amati dan catat menggunakan bahasa anda sendiri tampilan program setelah dijalankan. Perhatikan **cara pembacaan variabel** pada output program kemudian simpan pada Ms Word dan dikumpulkan

### Algoritma

Rumus konversi dari celcius ke reamur adalah  $R = 0.8 * C$ . Type data yang digunakan adalah **real**.

### Tabel konversi Dari Celsius ke

Skala yang diinginkan	Formula
Kelvin	$K = ^\circ C + 273,15$
Fahrenheit	$^{\circ}F = ^\circ C \times 1,8 + 32$
Reaumur	$^{\circ}R = ^\circ C \times 0,8$

### Program

```
var
Reamur, c : real;

begin
    Writeln('program konversi suhu ');
    Write('input suhu dalam celcius : ');
    ReadLn(c);

    reamur := 0.8 * c ;
    WriteLn('suhu dari celcius ke reamur : ', reamur:0:2);

    ReadLn();

end.
```

### Tugas

Buat konversi dari celcius ke kelvin dan fahrenheit

## 5. Algoritma Percabangan (Kondisional)

Algoritma percabangan adalah algoritma dimana terdapat baris yang akan diuji sebelum menjalankan baris berikutnya. Hasil dari pengujian kondisi adalah BOOLEAN atau kondisi yang bernilai benar dan salah.

### Tugas:

Amati dan catat menggunakan bahasa anda sendiri tampilan program setelah dijalankan. Perhatikan output program yang bertipe **BOOLEAN** atau **TRUE** dan **FALSE** kemudian simpan pada Ms Word dan dikumpulkan

### Contoh 1 :

```
var  
a, b, c : boolean;  
  
begin  
  a := 4 < 5;  
  b := 4 = 4;  
  a := 4 > 5;  
  
  Writeln(a);  
  Writeln(b);  
  Writeln(c);  
  
  ReadLn();  
  
end.
```

### Operator relasi yang mengembalikan hasil Boolean

Operator	Fungsi
:=	Menyatakan nilai
=	Sama dengan
<>	Tidak sama dengan
<	Lebih kecil
>	Lebih besar
<=	Lebih kecil atau sama dengan
>=	Lebih besar atau sama dengan

### IF Satu Kondisi

Bentuk umum dari IF satu kondisi dengan **satu kemungkinan** jawaban adalah:

**IF kondisi pengujian then**

Blok program yang dijalankan jika benar

Pada pengujian kondisi, IF akan menguji apakah kondisi pengujian bernilai TRUE atau FALSE. Jika TRUE maka baris program dibawah IF akan dijalankan.



**Contoh 2:**

```
var
nilai: integer;

begin
    Write('Input nilai yang akan diuji  : ');
    ReadLn(nilai);

    if nilai mod 2 = 0 then
        Write ('Nilai yang anda masukan adalah GENAP');

    ReadLn();

end.
```

Bentuk umum dari IF satu kondisi dengan **dua kemungkinan** jawaban adalah:

**IF kondisi pengujian then**

Blok program yang dijalankan jika pengujian bernilai BENAR

**Else**

Blok program yang dijalankan jika pengujian bernilai SALAH

Pada pengujian kondisi. IF akan menguji apakah kondisi pengujian bernilai TRUE atau FALSE. Jika TRUE maka baris program dibawah IF akan dijalankan. Jika FALSE maka baris program dibawah ELSE akan dijalankan.

**Contoh 3:**

```
var
nilai: integer;

begin
    Write('Input nilai yang akan diuji  : ');
    ReadLn(nilai);

    if nilai mod 2 = 0 then
        Write ('Nilai yang anda masukan adalah GENAP')

    else
        Write ('Nilai yang anda masukan adalah GANJIL');

    ReadLn();

end.
```

**Contoh 4 :**

Program Diskon pembelian. Jika pembelian lebih besar sama dengan 1000 maka akan mendapatkan diskon 20%.

**Algoritma**

Setiap pembelian akan diuji apakah nilai pembeliannya lebih besar dari 1000 maka pembeli akan mendapatkan diskon 20%, jika kurang dari 100 maka pembelian tetap sesuai dengan harga beli. Variabel yang digunakan adalah **real**

### Program

```
var
beli, total : real;

begin
  Write('Input pembelian : ');
  ReadLn(beli);

  if beli >= 1000 then
    begin
      total := beli * 0.8 ;
      Write ('total pembelian setelah diskon ', total:0:2);
    end
  else
    Write ('pembelian tanpa diskon ', beli:0:2);

  ReadLn();

end.
```

### IF Dua Kondisi

Bentuk umum dari IF dua kondisi dengan **satu kemungkinan** jawaban adalah:

**IF kondisi pengujian 1 and atau or kondisi pengujian 2 then**  
Blok program yang dijalankan jika pengujian bernilai BENAR  
**Else**  
Blok program yang dijalankan jika pengujian bernilai SALAH

Pada pengujian kondisi. IF akan menguji apakah kondisi pengujian bernilai TRUE atau FALSE. Jika TRUE maka baris program dibawah IF akan dijalankan.

Dua kondisi yang sering digunakan adalah AND dan OR. Tabel kebenaran AND dan OR adalah sebagai berikut:

**Tabel Kebenaran AND**

Kondisi 1	Kondisi 2	Output
True (1)	True (1)	True (1)
True (1)	False (0)	False (0)
False (0)	True (1)	False (0)
False (0)	False (0)	False (0)

**Tabel Kebenaran OR**

Kondisi 1	Kondisi 2	Output
True (1)	True (1)	True (1)
True (1)	False (0)	True (1)
False (0)	True (1)	True (1)
False (0)	False (0)	False (0)

### Contoh 5:

Buat sebuah program untuk menyatakan lulus dan tidak lulus siswa. Jika siswa memperoleh nilai antara 60 – 100 maka siswa dinyatakan lulus.

#### Algoritma :

Nilai siswa akan diuji pada kondisi 1 terlebih dahulu, kemudian kondisi 2 diuji hasil dari kondisi 1 dan kondisi 2 akan di kombinasikan sesuai dengan tabel kebenaran. Jika menggunakan AND maka kedua kondisi tersebut harus bernilai TRUE untuk menghasilkan nilai TRUE

**Tugas:**

Amati dan catat menggunakan bahasa anda sendiri tampilan program setelah dijalankan. Perhatikan output program yang berisi **pengujian kondisi** kemudian simpan pada Ms Word dan dikumpulkan

**Program**

```
var
    nilai : integer;
begin
    Write('Input Nilai Siswa : ');
    ReadLn(nilai);

    if (nilai >= 60) and (nilai <= 100) then
        Write ('Anda LULUS')
    else
        write ('Anda TIDAK LULUS');

    ReadLn();
end.
```

**Contoh 6:**

**Nested IF (IF bersarang)**

IF bersarang adalah kondisi IF yang dijalankan jika kondisi bernilai TRUE kemudian terjadi pengujian lagi. Jika kondisi bernilai FALSE maka IF didalam tidak akan dijalankan melainkan langsung ke Else.

**Algoritma:**

Berikut adalah contoh penggunaan IF bersarang. Pertama nilai akan diuji apakah nilai berada diantara 0 sampai 100 (IF pertama), jika kondisi ini bernilai TRUE maka akan menjalankan IF didalam (IF kedua) dimana nilai akan diuji apakah nilai yang dimasukan >= 60 dan <= 100 jika TRUE maka akan menampilkan 'Anda Lulus' jika FALSE maka akan menampilkan 'Anda TIDAK LULUS'. Tapi jika nilai tidak berada pada 0 sampai 100 atau tidak memenuhi IF pertama maka akan menampilkan tulisan 'Out of range'.

**Tugas:**

Amati dan catat menggunakan bahasa anda sendiri tampilan program setelah dijalankan. Perhatikan output program yang berisi **pengujian kondisi** kemudian simpan pada Ms Word dan dikumpulkan

```
var
    nilai : integer;
begin
    Write('Input Nilai Siswa : ');
    ReadLn(nilai);

    if (nilai >= 0) and (nilai <= 100) then
        begin
            if (nilai >= 60) and (nilai <= 100) then
                Write ('Anda LULUS')
            else
                write ('Anda TIDAK LULUS');
            end
        else
            write ('out of range');

        ReadLn();
    end.
```

**Tugas:**

Amati dan catat menggunakan bahasa anda sendiri tampilan program setelah dijalankan. Perhatikan output program yang berisi **pengujian kondisi** kemudian simpan pada Ms Word dan dikumpulkan

**Contoh 7:**

```
var
    nilai : integer;

begin
    Write('input nilai anda  : ');
    ReadLn(nilai);

    if (nilai >= 0) and (nilai <= 40 ) then
        Write('Nilai anda adalah D');

    if (nilai >= 41) and (nilai <= 55 ) then
        Write('Nilai anda adalah C');

    if (nilai >= 56) and (nilai <= 65 ) then
        Write('Nilai anda adalah B');

    if (nilai >= 66) and (nilai <= 100 ) then
        Write('Nilai anda adalah A')

    else
        Write('out of range');

    ReadLn();

end.
```

**Contoh 8:**

Program pengujian bilangan prima, genap dan ganjil.

**Tugas:**

Amati dan catat menggunakan bahasa anda sendiri tampilan program setelah dijalankan. Perhatikan output program yang berisi **pengujian kondisi** kemudian simpan pada Ms Word dan dikumpulkan

```
var
    bil: integer;

begin
    Write('input bilangan : ');
    ReadLn(bil);

    if ((bil=2) or (bil=3) or (bil=5) or (bil=7)) then
        Write ('bilangan prima dan ')
    else if ((bil mod 2 <> 0) and (bil mod 3 <> 0)
        and (bil mod 5 <> 0) and (bil mod 7 <> 0)) then
        write ('bilangan prima dan ');
    if bil mod 2 = 0 then
        Write ('bilangan genap')
    else
        write ('bilangan ganjil');

    ReadLn();

end.
```

**Tugas:**

Buat sebuah program untuk menampilkan Zodiak anda. Tanggal dan bulan dimasukkan menggunakan angka. Misalnya bulan 5 akan dianggap bulan mei.

**Case Of**

Case adalah percabangan yang mempunyai pengujian kondisi dimana variabel yang diuji hanya bertipe INTEGER dan CHAR.

Bentuk penulisan case of adalah:

**Case** {variable of type: *integer or character ONLY*} **of**

    {input statement- within inverted commas if of type **char**} : {code..}

    {input statement- within inverted commas if of type **char**} : {code..}

**End;** {End Case}

**Contoh 9:**

Berikut adalah sebuah program yang akan mengkonversikan angka menjadi huruf dari 0 sampai 9. saat kita memasukkan angka maka angka yang kita masukan akan diuji di sisi kiri jika angka yang kita masukan ada maka tampilan di sebelah kanan akan dijalankan.

**Program.**

```
var
    angka: integer;

begin
    Write('input angka : ');
    ReadLn(angka);
    case angka of
        0 : write ('NOL');
        1 : write ('SATU');
        2 : write ('DUA');
        3 : write ('TIGA');
        4 : write ('EMPAT');
        5 : write ('LIMA');
        6 : write ('ENAM');
        7 : write ('TUJUH');
        8 : write ('DELAPAN');
        9 : write ('SEMBILAN');
    else ;
        write ('OUT OF RANGE');
    end;

    ReadLn();
end.
```

**Contoh 10:**

Pengujian nilai yang dikonversikan menjadi huruf dimana nilai 0 – 100 akan dikonversikan menjadi A – D.

**Tugas:**

Amati dan catat menggunakan bahasa anda sendiri tampilan program setelah dijalankan. Perhatikan output program yang berisi ***pengujian kondisi*** kemudian simpan pada Ms Word dan dikumpulkan

**Program**

```
var
    nilai : integer;

begin
    Write('input nilai anda  : ');
    ReadLn(nilai);

    case nilai of
        0..40 :
            begin
                Write('Nilai anda adalah D');
            end;
        41..55 :
            begin
                Write('Nilai anda adalah C');
            end;
        56..65 :
            begin
                Write('Nilai anda adalah B');
            end;
        66..100 :
            begin
                Write('Nilai anda adalah A');
            end;
        else
            Write('out of range');
            end;
        ReadLn();
    end.
```

## Algoritma Pengulangan

Pengulangan adalah algoritma dimana baris program yang ditentukan akan di eksekusi secara berulang – ulang sampai batas yang telah ditentukan. Jika batas telah tercapai maka pengulangan akan berhenti.

### Contoh:

Jika kita mengupas 15 jeruk. Cara yang pertama kita lakukan adalah mengambil jeruk yang pertama kemudian mengupasnya. Setelah selesai jeruk yang pertama kemudian ke jeruk yang kedua dan seterusnya sampai jeruk ke 15 terkupas.  
dari cerita diatas terdapat 3 hal yang harus diketahui sebelum menggunakan pengulangan.

1. **Inisialisasi atau pengenalan objek** yang akan diulang. Contoh diatas adalah 15 jeruk yang belum terkupas.
2. **Iterasi atau pengulangan.** Contoh diatas adalah mengupas dari **1** sampai **15** kali untuk 15 jeruk.
3. **Kondisi berhenti.** adalah kondisi dimana semua proses pengulangan akan berhenti. dari contoh diatas adalah 15 jeruk yang telah terkupas.

## For Do

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

### 1. Perulangan positif.

Perulangan positif adalah perulangan dengan penghitung dari kecil ke besar atau pertambahan positif. Perulangan positif dapat dibentuk dengan menggunakan pernyataan For-To-Do.

Bentuk umum for do:

For **variabel-kontrol** := **nilai awal** To **nilai akhir** Do  
Statemen atau Pernyataan;

Variabel kontrol, nilai awal, nilai akhir harus betipe integer

### Contoh 1 :

Buat tampilan program yang akan menampilkan tulisan "Saya Belajar" sebanyak 5 baris.

### Program

```
var
  i : integer;

begin
  for i := 1 to 5 do
    Writeln('saya belajar pascal ');

  ReadLn();
end.
```

### Tugas:

Amati dan catat menggunakan bahasa anda sendiri tampilan program setelah dijalankan. Perhatikan output program yang berisi ***pengulangan*** kemudian simpan pada Ms Word dan dikumpulkan

**Contoh 2:**

```
var
  i : integer;

begin
  for i := 1 to 5 do
    Writeln('saya belajar pascal ');
    Writeln('di SMKN2 ');
    Writeln('buduran sidoarjo ');
  ReadLn();
end.
```

**Contoh 3:**

Jika baris yang akan dikerjakan oleh for lebih dari satu baris maka baris – baris yang akan dijalankan harus dimulai dengan begin dan di akhiri dengan end.

```
var
  i : integer;

begin
  for i := 1 to 5 do
    begin
      Writeln('saya belajar pascal ');
      Writeln('di SMKN2 ');
      Writeln('buduran sidoarjo ');
      WriteLn();
    end;
  ReadLn();
end.
```

**Contoh 3:**

Program untuk menampilkan angka pengulangan mulai 1 sampai sepuluh.

```
var
  i: integer;

begin
  for i := 1 to 10 do
    Write(i);

    ReadLn();
  end.
```

**Contoh 4:**

```
var
  i: integer;

begin
  for i := 1 to 10 do
    Write(i:4);

    ReadLn();
  End.
```



### Contoh 5:

Program yang digunakan untuk menampilkan bilangan prima antara 0 sampai 30.

#### Tugas:

Amati dan catat menggunakan bahasa anda sendiri tampilan program setelah dijalankan. Perhatikan output program yang berisi bilangan prima, ***buat contoh pengujian dengan beberapa angka.*** kemudian simpan pada Ms Word dan dikumpulkan

```
var
  i : integer;
begin
  for i := 2 to 30 do
    begin
      if ((i=2) or (i=3) or (i=5) or (i=7)) then
        Write (i:3)
      else if ((i mod 2 <> 0) and (i mod 3 <> 0)
        and (i mod 5 <> 0) and (i mod 7 <> 0)) then
        write (i:3);
      end;
    end;
  ReadLn();
end.
```

### Perulangan tersarang

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

### Contoh 6:

```
var
  i,j: integer;
begin
  for i := 1 to 10 do
    begin
      for j := 1 to 5 do
        begin
          Write(i:8);
          Write(j:3);
        end;
        WriteLn();
      end;
    end;
  ReadLn();
end.
```

### Perulangan negatif

Perulangan negatif adalah perulangan dengan penghitung dari besar ke kecil atau pertambahan negatif. Perulangan negatif dapat dibentuk dengan menggunakan pernyataan For-DownTo-Do.

Bentuk umum:

For **variabel-kontrol** := **nilai awal** DownTo **nilai akhir** Do  
Statemen atau pernyataan

**Contoh 7:**

Program untuk menampilkan bilangan yang mulai dari 10 turun sampai ke 0

**program**

```
var
    i : integer;
begin
    for i := 10 downto 0 do
        write (i:3);

        readln;
    end.
```

**Contoh 8:**

Program untuk menampilkan bintang dalam formasi segitiga.

**Tugas:**

Amati dan catat menggunakan bahasa anda sendiri tampilan program setelah dijalankan. Perhatikan output program yang menampilkan ***bintang dalam formasi segitiga***. kemudian simpan pada Ms Word dan dikumpulkan

**Program:**

```
var
    i, j : integer;
begin
    for i := 10 downto 0 do
        begin
            for j := 1 to i do
                begin
                    write ('*');
                end;
                WriteLn();
            end;

            readln;
        end.
```

## While Do

Perulangan ini menggunakan pernyataan While—Do. Pernyataan While—Do digunakan untuk melakukan proses perulangan suatu pernyataan atau blok pernyataan terus-menerus selama kondisi ungkapan logika pada While masih bernilai logika benar.

Bentuk Umum

While <b>kondisi</b> Do <b>Pernyataan</b>
---

### Contoh 1:

**Baris i:= 0** adalah baris yang menyatakan awal pengulangan. Dan **while i < 10** adalah baris menyatakan akhir pengulangan. Baris **i:= i + 1** adalah baris yang menyatakan increment atau pengulangan naik satu step.

### Program

```
var
  i : integer;

begin
  i := 0;
  while i < 10 do
  begin
    Writeln('looping ke : ',i, ' saya belajar pascal');
    i:= i + 1 ;
  end;
  ReadLn();
end.
```

Program yang digunakan untuk menampilkan bilangan prima antara 0 sampai 100. menggunakan while do.

### Tugas:

Amati dan catat menggunakan bahasa anda sendiri tampilan program setelah dijalankan. Perhatikan output program yang berisi bilangan prima, **catat perbedaan menggunakan for dan while**. kemudian simpan pada Ms Word dan dikumpulkan

### Contoh 2:

```
var
  i : integer;

begin
  i := 2 ;
  while i < 100 do
  begin
    if ((i=2) or (i=3) or (i=5) or (i=7)) then
      Write (i:3)
    else if ((i mod 2 <> 0) and (i mod 3 <> 0)
      and (i mod 5 <> 0) and (i mod 7 <> 0)) then
      write (i:3);

    i := i + 1 ;
  end;

  ReadLn();
end.
```

## Repeat Until

Struktur Repeat...Until digunakan untuk mengulang pernyataan-pernyataan atau blok pernyataan sampai kondisi yang diseleksi di Until terpenuhi.

Bentuk umumnya adalah:

Repeat **pernyataan** Until **kondisi akhir**

### Contoh 1:

Baris **i:= 0** adalah baris yang menyatakan awal pengulangan. Dan **until = 10** adalah baris menyatakan akhir pengulangan. Baris **i:= i + 1** adalah baris yang menyatakan increment atau pengulangan naik satu step.

```
var
  i : integer;

begin
  i := 0 ;
  repeat
    WriteLn('looping ke : ',i,' saya belajar pascal');
    i := i + 1 ;
  until i = 10 ;
  ReadLn();
end.
```

### Contoh 2:

Program yang digunakan untuk menampilkan bilangan prima antara 0 sampai 100. menggunakan repeat until.

#### Tugas:

Amati dan catat menggunakan bahasa anda sendiri tampilan program setelah dijalankan. Perhatikan output program yang berisi bilangan prima, **catat perbedaan menggunakan for, while, dan repeat**. kemudian simpan pada Ms Word dan dikumpulkan

```
var
  i : integer;

begin
  i := 2 ;
  repeat
    begin
      if ((i=2) or (i=3) or (i=5) or (i=7)) then
        Write (i:3)
      else if ((i mod 2 <> 0) and (i mod 3 <> 0)
        and (i mod 5 <> 0) and (i mod 7 <> 0)) then
        write (i:3);

      i := i + 1 ;
    end;
  until i = 100 ;

  ReadLn();
end.
```

**Contoh 3:**

**Membuat deret fibonacci:**

Deret fibonacci diawali dari deret 2 angka

1	1
---	---

Angka ini kemudian dijumlahkan menjadi angka yang ke 3

1	1	2
---	---	---

Setiap angka berikutnya adalah penjumlahan dari 2 angka sebelumnya.

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

Demikian sampai jumlah deret yang di inginkan ditampilkan. Contoh diatas adalah 8 deret fibonacci.

```
var
    Fibonacci1, Fibonacci2 : integer;
    temp : integer;
    count,n : integer;

begin
    writeln ('----MEMBUAT DERET FIBONACCI ---');
    write ('masukkan jumlah deret : ');
    readln (n);
    count := 0;
    Fibonacci1 := 0;
    Fibonacci2 := 1;
    repeat
        write (Fibonacci2:3);
        temp := Fibonacci2;
        Fibonacci2 := Fibonacci1 + Fibonacci2;
        Fibonacci1 := Temp;
        count := count + 1
    until count = n;

    readln;
end.
```

## Operasi String

Suatu string di dalam Pascal dapat dioperasikan dengan berbagai macam tujuan. Beberapa string dapat dirangkai menjadi satu, dapat digunakan sebagai kondisi yang diseleksi dan lain sebagainya. Pascal menyediakan beberapa prosedur standar dan fungsi standar untuk operasi string.

### Merangkai String

String hanya mempunyai sebuah operator, yaitu operator `+'. Bila operator ini digunakan untuk penambahan nilai numerik, maka akan berfungsi menjumlahkan dua buah nilai elemen numerik. Sedang pada string, operator ini digunakan untuk merangkai dua buah elemen string menjadi sebuah string. Panjang maksimum suatu string yang diijinkan oleh Pascal adalah 255 karakter.

```
Const
  Tulisan1 ='Saya';
  Tulisan2 = 'Belajar';
  Tulisan3 = 'Pascal';
Var
  Kalimat:string [200];
Begin
  Kalimat:=tulisan1+' '+tulisan2+' ' + tulisan3;
  Write(kalimat);
  readln;
End.
```

### Penjelasan

```
Kalimat:=tulisan1+' '+tulisan2+' ' + tulisan3;
```

Deklarasi **const** digunakan untuk menetapkan tulisan1 dengan "Bahasa", tulisan2 dengan "turbo" dan tulisan 3 dengan "pascal". Program akan mengerjakan operasi sesuai dengan nilai yang telah ditetapkan atau konstanta (**Const**)

Pada hasil penambahan kalimat kita menambahkan tulisan1 ditambah (+) dengan spasi (' ') kemudian ditambahkan dengan tulisan2 kemudian ditambahkan dengan spasi (dua tanda petik satu dimana diantara tanda petik di beri jarak ' ') kemudian ditambah dengan tulisan3. Hasil dari penjumlahan akan terlihat pada tampilan program.

## Array (Larik)

### Definsi Larik

Suatu larik (array) adalah tipe terstruktur yang terdiri dari sejumlah komponen-komponen yang mempunyai tipe sama. Suatu larik mempunyai jumlah komponen yang banyaknya tetap. Banyaknya komponen dalam suatu larik ditunjukkan oleh suatu indeks yang disebut tipe indeks. Tipe indeks ini berbentuk ungkapan tipe ordinal (tipe data yang berurutan). Tiap-tiap komponen pada larik dapat diakses dengan menunjukkan nilai indeksnya.

### Contoh:

Kita mempunyai:

larik bunga yang berisi data {mawar, melati, anggrek}

larik hari yang berisi data {senin, Selasa, Rabu, Kamis, Jumat, Sabtu, Minggu}

### Array Satu Dimensi

Larik ini memiliki data yang berurutan dengan satu jenis yang sama dan tidak berbentuk tabel.

Dari contoh diatas dapat di gambarkan:

**Larik bunga**

mawar	melati	Anggrek
Indeks 1	Indeks 2	Indeks 3

Dari gambar diatas jika kita ingin menampilkan bunga melati maka kita cukup memanggil nomer indeks nya saja.

Bentuk umum array satu dimensi adalah:

**Nama larik** : array [jumlah indeks] of tipe data

**Deklarasi array**

Dari contoh bunga diatas dapat di deklarasikan array sebagai berikut:

```
var
    bunga : array[1..3] of string;
begin
    bunga [1]:='mawar';
    bunga [2]:='melati';
    bunga [3]:='anggrek';
    write (bunga [1]);
    readln;
end.
```

```
var
    bunga : array[1..3] of string;
    x : integer;
begin
    bunga [1]:='mawar';
    bunga [2]:='melati';
    bunga [3]:='anggrek';
    write ('Masukkan Pilihan anda      :');
    readln (x);
    writeln ('bunga yang anda pilih adalah      :',bunga [x]);

    readln;
end.
```

```
var
    bunga : array[1..10] of string;
    i : integer ;
begin
    for i := 1 to 3 do
        begin
            Write('input nama bunga : ');
            ReadLn(bunga[i]);
        end;
        WriteLn();
        writeln ('bunga yang anda masukan adalah : ');
        for i := 1 to 3 do
            begin
                Writeln('bunga ke : ',i,' ', bunga[i]);
            end;
        readln;
    end.
```