

8

KONSEP DASAR STRUKTUR DATA & REVIEW ARRAY

ALGORITMA DAN STRUKTUR DATA

Struktur is?

- Koleksi dari variabel yang dinyatakan dengan sebuah nama.
- Dipakai untuk mengelompokkan beberapa informasi yang berkaitan menjadi sebuah satu kesatuan.

Mendeklarasikan Struktur


- ▶ TIPE STRUKTUR
- ▶ struct data_tanggal
- ▶ {
- ▶ int tanggal;
- ▶ int bulan; int tahun;
- ▶ };

PENDEKLARASIAN STRUKTUR

```
struct data_tanggal  
{  
    int tanggal;  
    int bulan;  
    int tahun;  
} tgl_lahir;
```

BENTUK UMUM MENDEFINISIKAN DAN MENDEKLARASIKAN STRUKTUR

```
{  
tipe field1;  
tipe field2;  
.  
.  
tipe fieldn;  
}variabel_struktur1, ... , variabel_strukturM;
```



KONSEP DASAR

- **Struktur data** adalah cara penyimpanan, penyusunan dan pengaturan data di dalam media penyimpanan komputer sehingga data tersebut dapat digunakan secara efisien, termasuk operasi.-operasi di dalamnya

Tiga bagian utama yaitu :

- Himpunan struktur dari tempat penyimpanan atau storage.
Merupakan koleksi dari variable dan hubungan antara satu variable dengan variable yang lain.
- Himpunan dari fungsi-fungsi dasar.
Dapat digunakan pada struktur tempat penyimpanan yang ada dan dapat digunakan pada setiap bagian dari program
- Himpunan dan algoritma digunakan untuk pengubahan dari struktur tempat penyimpanan.

- Array (larik) : sebuah variabel yang dapat menyimpan lebih dari satu nilai sejenis (memiliki tipe data sama).

A[1]	A[2]	A[3]	A[4]	A[5]	A[6]	A[7]	A[8]	A[9]	A[10]
1	2	3	4	5	6	7	8	9	10
10	20	30	40	50	60	70	80	90	100

Nilai elemen array
Indeks array
Elemen array

- r dan akan menempati alamat memori yang berbeda,
- array tersebut memiliki ukuran 40 byte, yang berasal dari 10 x 4.

DEKLARASI ARRAY

- Array dideklarasikan di dalam bagian deklarasi variabel.
- Array juga akan dideklarasikan di bagian definisi tipe (di bawah kata kunci `type`).
- Pendeklarasian array dilakukan dengan menggunakan kata kunci `array` dan tipe data yang akan disimpan di dalamnya. (PASCAL)
- Batas-batas indeksinya diapit oleh tanda *bracket* (`[]`).

```
NamaArray : array [Indeks Awal .. IndeksAkhir] of tipe data;
```

CONTOH

- array dengan nama A yang berisi 10 buah elemen bertipe integer

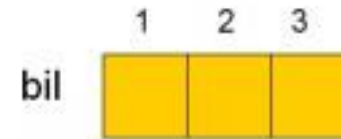
► Pada

```
Var  
A : array [1 .. 10] of integer;
```


Manipulasi array

NamaArray [indeks] := nilai;

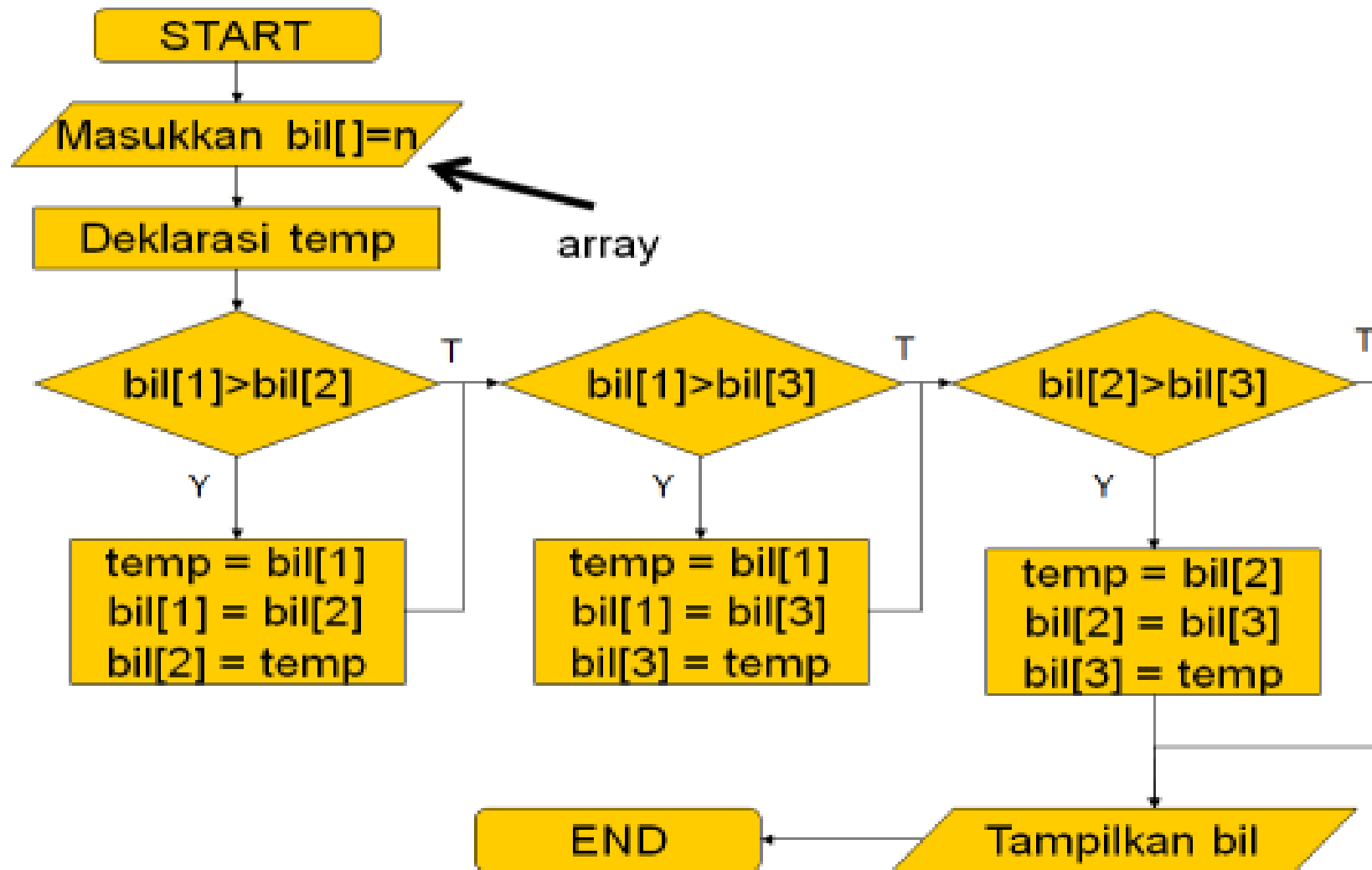
- Contoh mengakses array



Jika (`bil[2] < bil[1]`)
`temp = bil[1]`
`bil[1] = bil[2]`
`bil[2] = temp`

Jika (`bil[3] < bil[1]`)
`temp = bil[1]`
`bil[1] = bil[3]`
`bil[3] = temp`

Jika (`bil[3] < bil[2]`)
`temp = bil[2]`
`bil[2] = bil[3]`
`bil[3] = temp`



membuat program untuk menyimpan sekumpulan data?

```
Var |  
N1, n2, n3, n4, n5, n6, n7, n8, n9, n10 : real;  
Begin  
Writeln('masukkan data ke-1 : '); readln(n1);  
Writeln('masukkan data ke-2 : '); readln(n2);  
Writeln('masukkan data ke-3 : '); readln(n3);  
Writeln('masukkan data ke-4 : '); readln(n4);  
Writeln('masukkan data ke-5 : '); readln(n5);  
Writeln('masukkan data ke-6 : '); readln(n6);  
Writeln('masukkan data ke-7 : '); readln(n7);  
Writeln('masukkan data ke-8 : '); readln(n8);  
Writeln('masukkan data ke-9 : '); readln(n9);  
Writeln('masukkan data ke-10 : '); readln(n10);  
End.
```

BAGAIMANA JIKA LEBIH DARI 10? BAHKAN RIBUAN?

```
Const max = 10;  
Var  
n : array [1 .. max] of real;  
i : integer;  
begin  
  for i:= 1 to max do  
    writeln('Masukkan data ke-', i, ' : '); readln(n[i]);  
  end.
```

Array Sebagai Tipe Data Bentukan

Type

Bilangan = array [1 .. 100] of integer;

Vokal = array [1 .. 5] of char;

Var

X : bilangan;

Vowel : vokal;

contoh kode yang menunjukkan penggunaan variabel–variabel tersebut

```
Begin  
X[1] := 1;  
X[2] := 2;  
...  
Vowel [1] := 'a';  
Vowel [2] := 'i';  
...  
End.
```

Array Konstan

Const

```
NamaArray : array [indexAwal .. indeksAkhir] of tipe_data =  
    Nilai1, nilai2, ...);
```

Const

```
A : array [1 .. 5] of char = ('A' , 'B' , 'C' , 'D' , 'E');
```

array A di atas bersifat konstan, maka kita tidak dapat menggantikan nilainya dengan nilai lain, seperti yang ditunjukkan oleh kode di bawah ini

```
A [1] := 'V' ; {SALAH, karena elemen A [1] selalu bernilai 'A'}
```

```
A [2] := 'W' ; {SALAH, karena elemen A [2] selalu bernilai 'B'}
```

```
A [1] := 'X' ; {SALAH, karena elemen A [3] selalu bernilai 'C'}
```

```
A [1] := 'Y' ; {SALAH, karena elemen A [4] selalu bernilai 'D'}
```

```
A [1] := 'Z' ; {SALAH, karena elemen A [5] selalu bernilai 'E'}
```


Array Sebagai Parameter

```
Type  
Bilangan = array[1..100] of integer;  
Procedure inputArray(a: bilangan; N: integer);  
Var  
  i : integer;  
Begin  
  For i := 1 to N do  
    Write ('masukkan elemen array ke-', i); readln(A[i]);  
End;
```



Terima
kasih