

PRAKTIKUM 2

VARIABEL, TIPE DATA DAN OPERATOR

2.1 TUJUAN PRAKTIKUM

Tujuan Umum

Mahasiswa dapat memahami konsep tipe data

Tujuan Khusus

Mahasiswa dapat :

1. Konsep Variabel dan Konstanta beserta cara deklarasi dan Inisialisasinya pada Algoritma.
2. Memahami berbagai macam tipe data.
3. Menerapkan konsep-konsep tersebut dengan menggunakan flowgorithm.

2.2 TEORI SINGKAT

Program komputer pada dasarnya memanipulasi objek (data) di dalam memori. Peubah (variabel) dan konstanta adalah objek data dasar yang dimanipulasi dalam program. Deklarasi daftar peubah digunakan untuk menyatakan tipe peubah. Operator menspesifikasikan operasi apa saja yang dapat dilakukan terhadap peubah dan konstansa. Ekspresi mengkombinasikan peubah-peubah dan konstanta untuk menghasilkan nilai baru.

2.2.1 Variabel

Variabel adalah suatu pengenal (identifier) yang digunakan untuk mewakili suatu nilai tertentu di dalam proses program. Berbeda dengan konstanta yang nilainya selalu tetap, nilai dari suatu variable bisa diubah-ubah sesuai kebutuhan. Nama dari suatu variabel dapat ditentukan sendiri oleh pemrogram dengan aturan sebagai berikut (merujuk pada bahasa pemrograman C):

- a. Terdiri dari gabungan huruf dan angka dengan karakter pertama harus berupa huruf. Dalam bahasa C variabel bersifat case-sensitive artinya huruf besar dan kecil dianggap berbeda. Jadi antara nim, NIM dan Nim dianggap berbeda.
- b. Tidak boleh mengandung spasi.
- c. Tidak boleh mengandung simbol-simbol khusus, kecuali garis bawah (underscore). Yang termasuk symbol khusus yang tidak diperbolehkan antara lain : \$, ?, %, #, !, &, *, (,), -, +, = dsb.
- d. Bukan merupakan identifier yang sudah ada, contoh : main, for, while, if, dsb.

Contoh penamaan variabel yang **benar** :

NIM, a, x, nama_mhs, f3098, f4, nilai, budi.

Contoh penamaan variable yang **salah** :

%nilai_mahasiswa, 80mahasiswa, rata-rata, ada spasi, penting!

Catatan : tidak seperti bahasa C, pada Flowgorithm tidak mengijinkan menggunakan karakter garis bawah (underscores).

2.2.2 Tipe Data

Tipe data merupakan bagian program yang paling penting karena tipe data mempengaruhi setiap instruksi yang akan dilaksanakan oleh komputer. Misalnya saja 5 dibagi 2 bisa saja menghasilkan hasil yang berbeda tergantung tipe datanya. Jika 5 dan 2 bertipe integer maka akan menghasilkan nilai 2, namun jika keduanya bertipe float maka akan menghasilkan nilai 2.5000000. Pemilihan tipe data yang tepat akan membuat proses operasi data menjadi lebih efisien dan efektif.

Tipe data dapat dikelompokkan menjadi dua macam, tipe data dasar (tipe data sederhana) dan tipe data bentukan (struktur data). tipe dasar adalah tipe yang dapat langsung dipaka, sedangkan tipe bentukan dibentuk dari tipe dasar atau tipe bentukan lain yang didefinisikan sebelumnya.

Contoh tipe data sederhana:

- ◆ Numerik, yang terdiri dari Numerik integer (bilangan bulat) dan Numerik real (bilangan riil)
- ◆ Karakter, yaitu : Alfabet (a .. z, A .. Z), Angka (0 .. 9), Simbol khusus (+ ? ' ! [] { } ... dll)
- ◆ Boolean (logika), yang terdiri dari True dan False

Contoh struktur data:

- ◆ Array (larik)
- ◆ String
- ◆ Record
- ◆ List (daftar)
- ◆ Tree

2.2.3 Operator

Operator Penugasan

Operator Penugasan (Assignment operator) dalam bahasa C berupa tanda sama dengan ("="). Contoh :

```
nilai = 80;
A = x * y;
```

Artinya : variable "nilai" diisi dengan 80 dan variable "A" diisi dengan hasil perkalian antara x dan y.

Operator Aritmatika

Bahasa C menyediakan lima operator aritmatika, yaitu :

- * : untuk perkalian
- / : untuk pembagian
- % : untuk sisa pembagian (modulus)
- + : untuk penambahan
- : untuk pengurangan

Catatan : operator % digunakan untuk mencari sisa pembagian antara dua bilangan. Misalnya :

```
9 % 2 = 1
9 % 3 = 0
9 % 5 = 4
9 % 6 = 3
```

Operator Hubungan (Perbandingan)

Operator Hubungan digunakan untuk membandingkan hubungan antara dua buah operand (sebuah nilai atau variable). Operator hubungan dalam bahasa C :

Operator	Arti	Contoh	
<	Kurang dari	x < y	Apakah x kurang dari y
<=	Kurang dari sama dengan	x <= y	Apakah x kurang dari sama dengan y
>	Lebih dari	x > y	Apakah x lebih dari y
>=	Lebih dari sama dengan	x >= y	Apakah x lebih dari sama dengan y
==	Sama dengan	x == y	Apakah x sama dengan y
!=	Tidak sama dengan	x != y	Apakah x tidak sama dengan y

Operator Logika

Jika operator hubungan membandingkan hubungan antara dua buah operand, maka operator logika digunakan untuk membandingkan logika hasil dari operator-operator hubungan. Operator logika ada tiga macam, yaitu :

&& : Logika AND (DAN)

- || : Logika OR (ATAU)
- ! : Logika NOT (INGKARAN)

Operator Unary

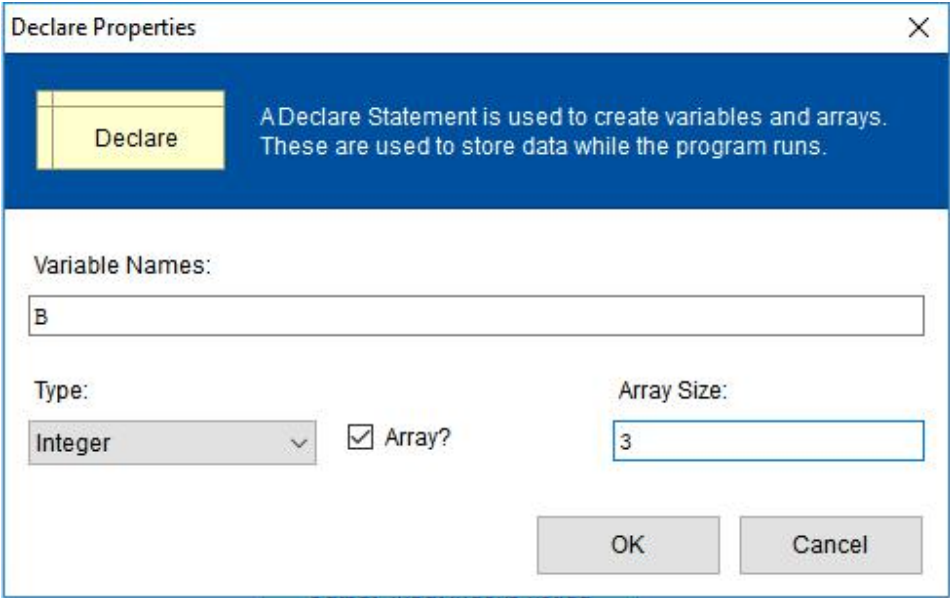
Operator Unary merupakan operator yang hanya membutuhkan satu operand saja. Dalam bahasa C terdapat beberapa operator unary, yaitu :

Operator	Arti/Maksud	Letak	Contoh	Equivalen
-	Unary minus	Sebelum operator	A + -B * C	A + (-B) * C
++	Peningkatan dengan penambahan nilai 1	Sebelum dan sesudah	A++	A = A + 1
--	Penurunan dengan pengurangan nilai 1	Sebelum dan sesudah	A--	A = A - 1
sizeof	Ukuran dari operand dalam byte	Sebelum	sizeof(I)	-
!	Unary NOT	Sebelum	!A	-
~	Bitwise NOT	Sebelum	~A	-
&	Menghasilkan alamat memori operand	Sebelum	&A	-
*	Menghasilkan nilai dari pointer	Sebelum	*A	-

2.3 PELAKSANAAN PRAKTIKUM

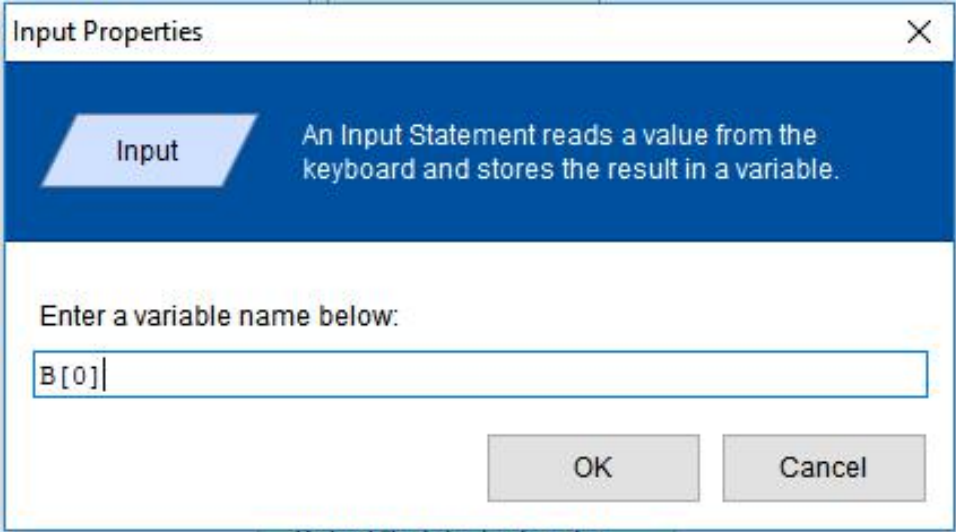
Praktikum 2.1

- 1. Buka flowgorithm, buat file baru.
- 2. Buatlah deklarasi variabel B bertipe data array integer dengan ukuran array 3.



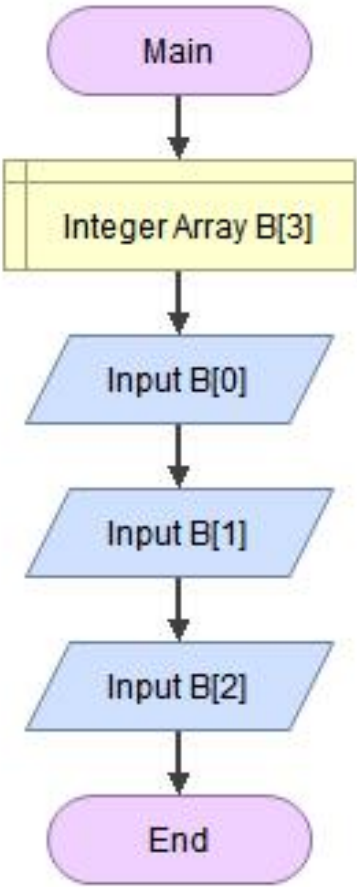
Gambar 2.1 Mendeklarasikan Array B

3. Tambahkan simbol input untuk memberi nilai indeks ke-“0” pada array B.



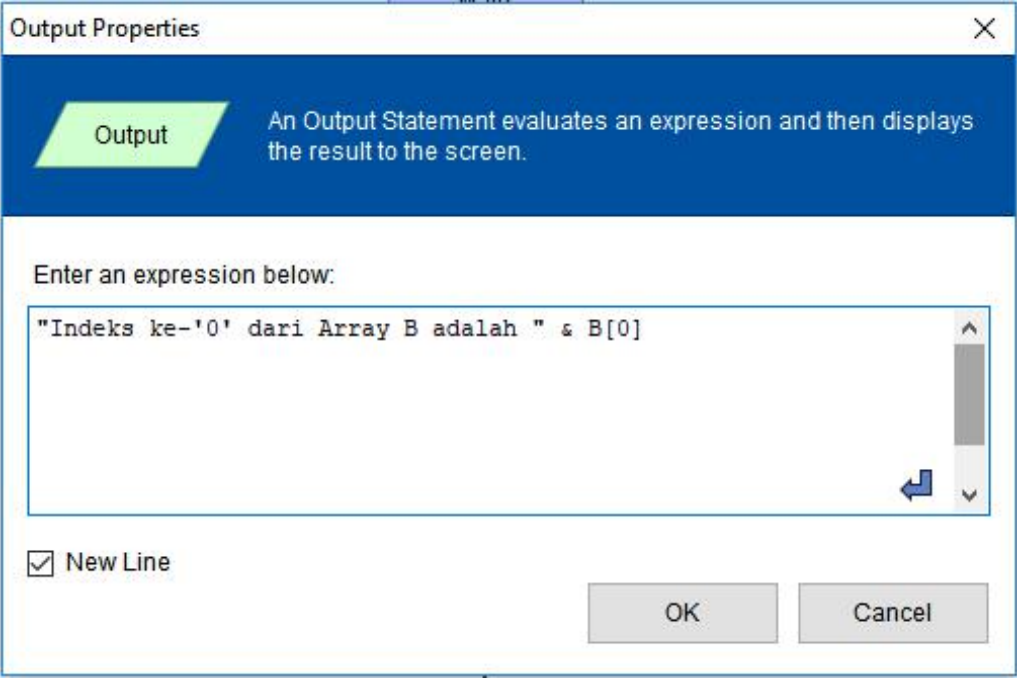
Gambar 2.2 Menambahkan simbol input untuk memberi nilai indeks ke-“0” pada array B

4. Tambahkan simbol input untuk tiap indeks array B.



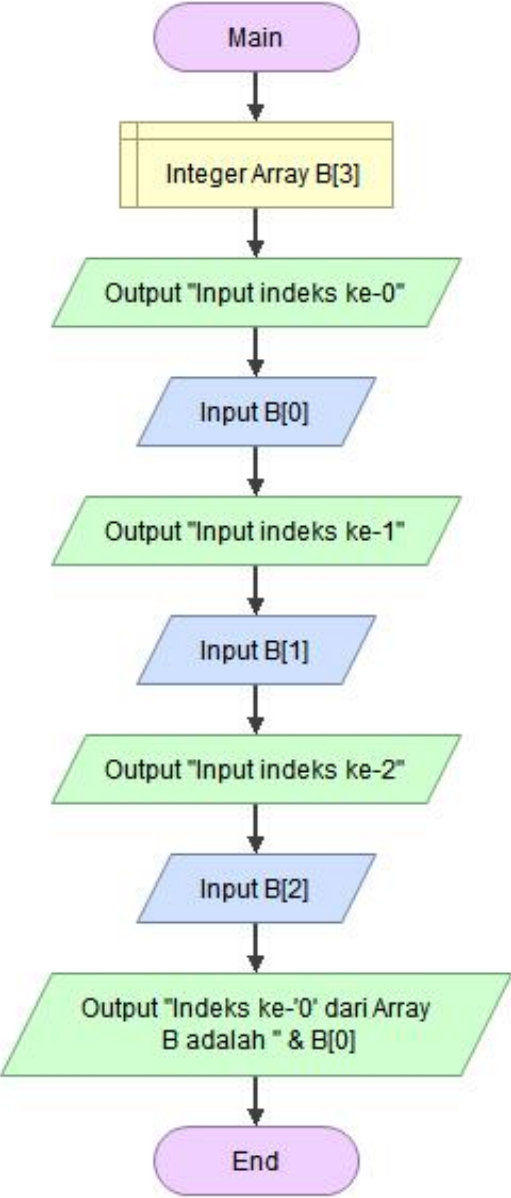
Gambar 2.3 Flowchart untuk menerima input tiap indeks array B

5. Tampilkan nilai salah satu indeks array B



Gambar 2.4 Menampilkan nilai salah satu indeks array B

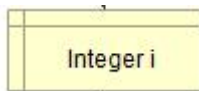
6. Lengkapi flowchart sehingga menjadi seperti gambar dibawah ini.



Gambar 2.5 Flowchart lengkap

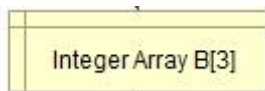
Praktikum 2.2

- 1. Buat file baru pada flowgorithm.
- 2. Deklarasikan variabel i dengan tipe data integer.



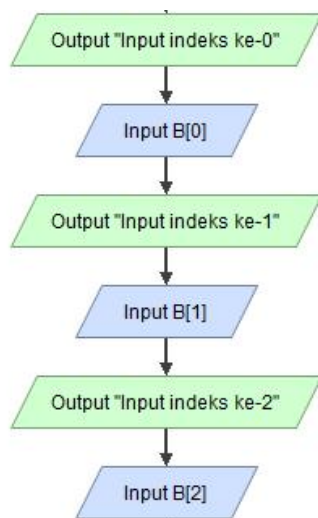
Gambar 2.6 Mendeklarasikan variabel i

- 3. Deklarasikan variabel B sebagai Array bertipe integer dengan panjang array 3.



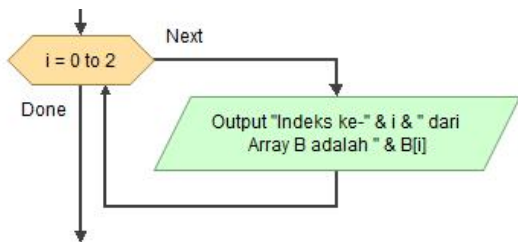
Gambar 2.7 Mendeklarasikan variabel B sebagai array integer berukuran 3

- 4. Tambahkan simbol output dan input untuk memberi nilai pada variabel setiap indeks Array B.

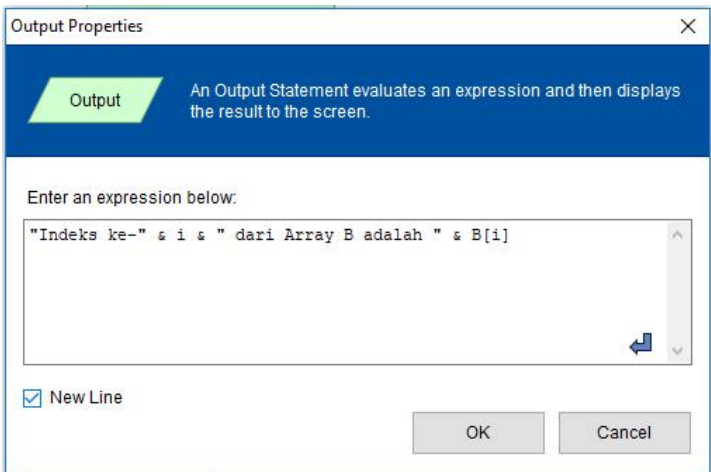


Gambar 2.8 Memberi nilai pada masing-masing indeks array B

- 5. Tampilkan semua isi Array B menggunakan loop "For".

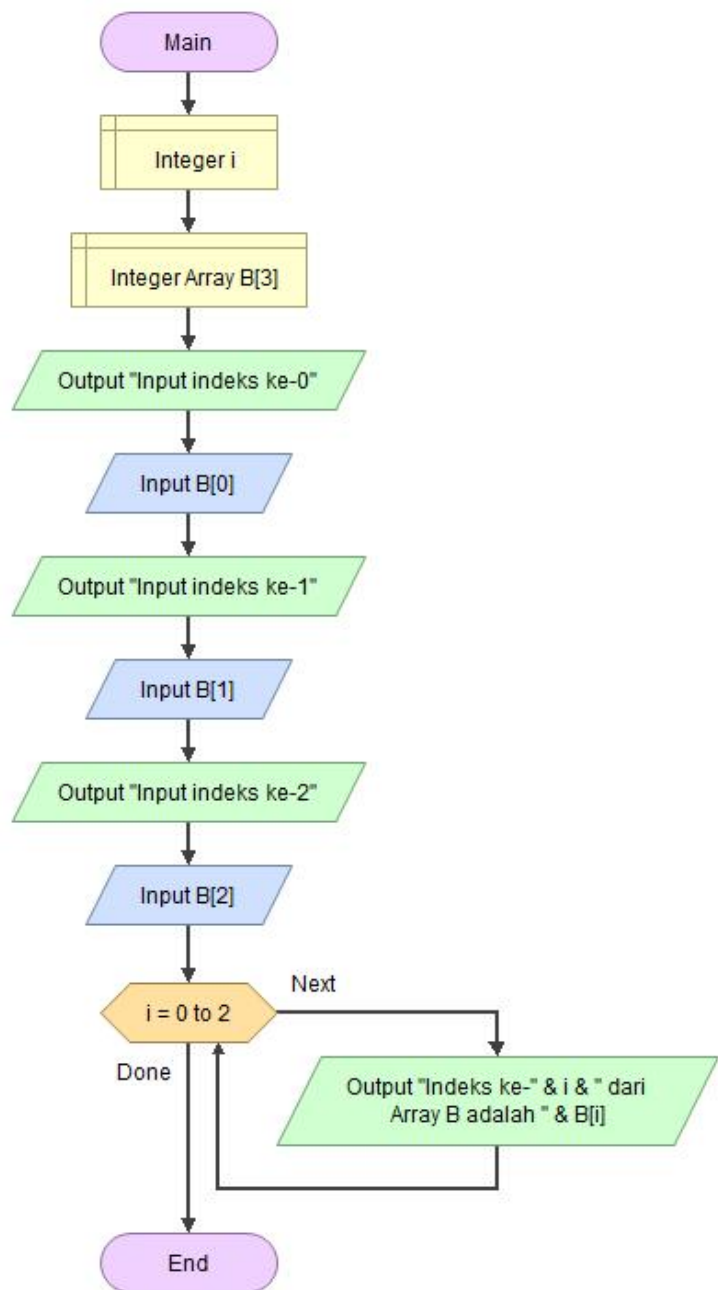


Gambar 2.9 Menampilkan seluruh isi array B menggunakan loop "For"



Gambar 2.10 Menambahkan simbol loop "For" pada flowchart

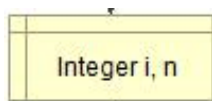
6. Berikut ini flowchart secara lengkap



Gambar 2.11 Flowchart lengkap praktikum 2.2

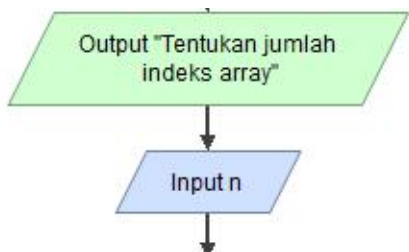
Praktikum 2.3

- 1. Buat file baru pada flowgorithm.
- 2. Deklarasikan variable i dan n dengan tipe data integer



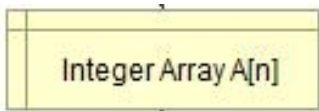
Gambar 2.12 Mendeklarasikan variabel i dan n

- 3. Tambahkan simbol output dan input untuk memberi nilai pada variabel n

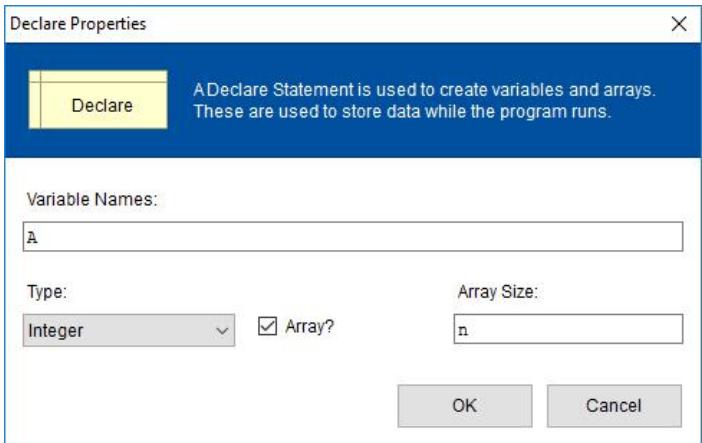


Gambar 2.13 Simbol output dan input untuk memberi nilai pada variabel n

4. Deklarasikan variabel A sebagai Array bertipe integer dengan panjang array “n”.

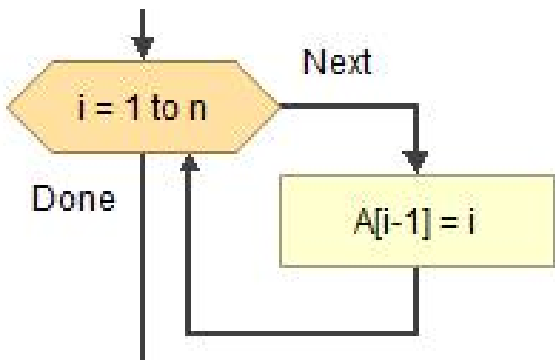


Gambar 2.14 Mendeklarasikan variabel A sebagai array integer berukuran n



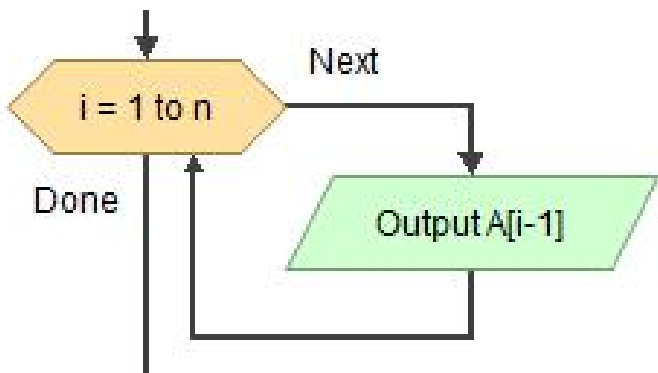
Gambar 2.15 Menambahkan simbol deklarasi untuk array A

5. Tambahkan loop “For” pada flowchart dengan start value “1” dan end value “n”. Assign tiap indeks array A dengan nilai dari variabel i.



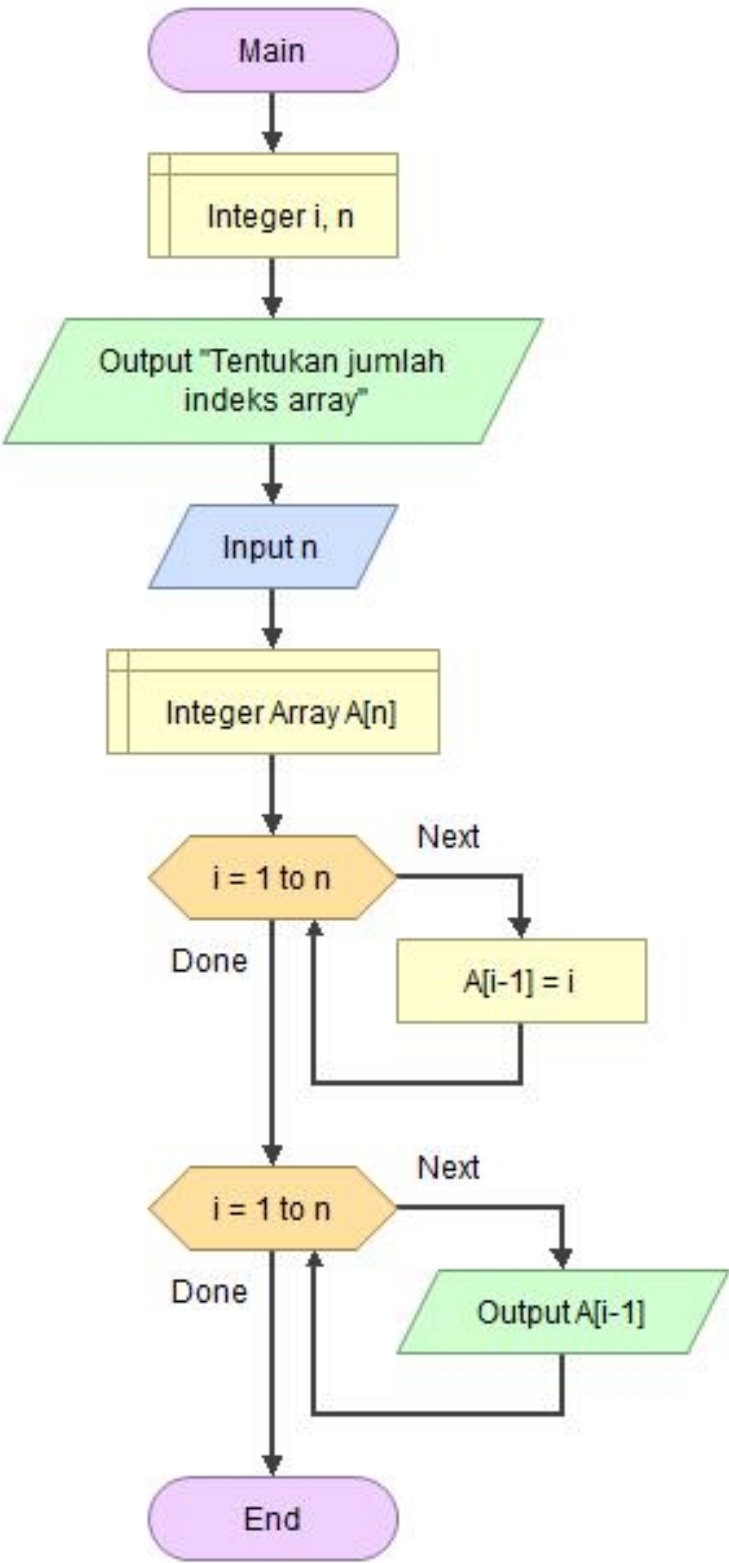
Gambar 2.16 Loop “For” untuk memberi nilai tiap indeks array A

6. Tampilkan tiap indeks array A menggunakan simbol loop “For”.



Gambar 2.17 Loop “For” untuk menampilkan seluruh isi array A

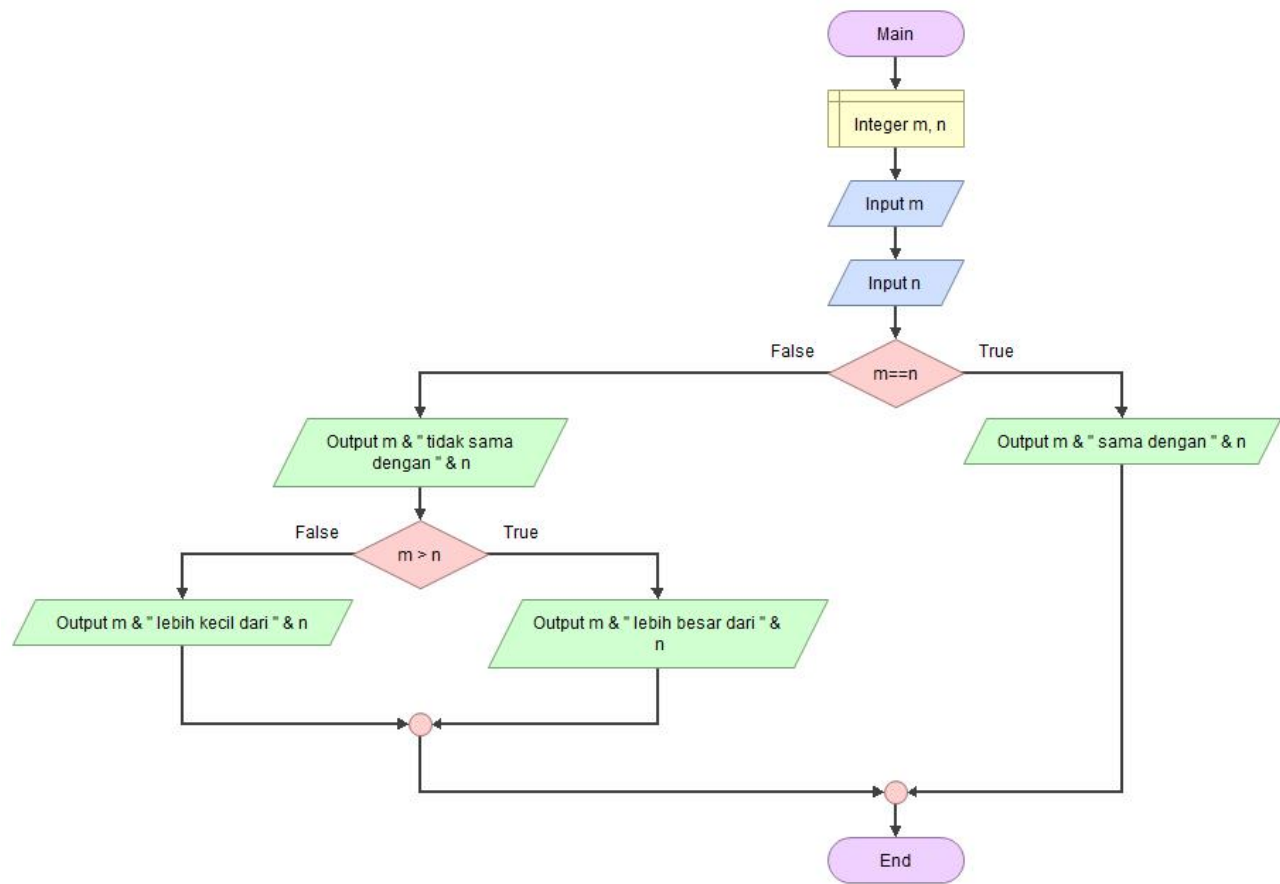
7. Berikut ini Flowchart secara lengkap



Gambar 2.18 Flowchart lengkap praktikum 2.3

Praktikum 2.4

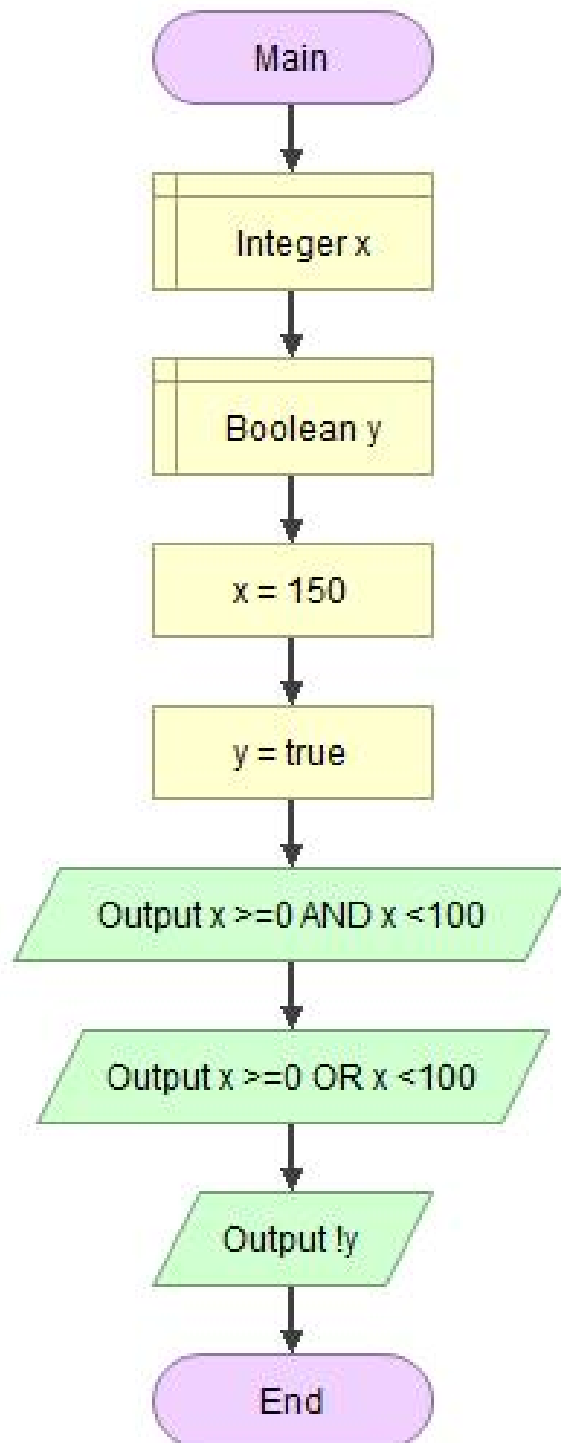
Buatlah flowchart seperti gambar dibawah ini.



Gambar 2.19 Flowchart lengkap praktikum 2.4

Praktikum 2.5

Buatlah flowchart seperti gambar dibawah ini.



Gambar 2.20 Flowchart lengkap praktikum 2.5

Latihan

Buatlah sebuah program menggunakan flowgorithm untuk :

1. Menghitung bonus karyawan dengan aturan sebagai berikut :

- Membaca input nama karyawan, lama jam lembur, lama jam kerja
- Menghitung bonus berdasarkan tabel dibawah ini

Aturan Bonus	
$JLEMBUR + (2/3)*JKERJA$	Bonus
>40 jam	Rp. 500.000
$30 < x \leq 40$ jam	Rp. 400.000
$20 < x \leq 30$ jam	Rp. 300.000
$10 < x \leq 20$ jam	Rp. 200.000
≤ 10 jam	Rp. 100.000

- Menampilkan output berupa nama karyawan disertai dengan lama jam lembur, lama jam kerja dan bonus yang didapat
2. Menentukan biaya parkir yang dihitung berdasarkan lama parkir. Lama parkir dihitung dari selisih jam masuk dan jam keluar yang di-input. Biaya parkir 2 jam pertama 2000, perjam berikutnya 500.
3. Menghitung rata-rata dari sejumlah nilai yang diinput (gunakan array dan perulangan)!