

AGUSTUS, 2020



PEMROGRAMAN DASAR

STRUKTUR DASAR PEMROGRAMAN [FLOWCHART, VARIABLE,
OPERATOR]

MODUL 1

TIM PENYUSUN:

- FAUZI DWI SETIAWAN SUMADI ST., M.CompSc (INSTRUKTUR)
- ALIF SYIFA ARSYILA (ASISTEN)

PRESENTED BY: LAB. TEKNIK INFORMATIKA
UNIVERSITAS MUHAMMADIYAH MALANG

PEMROGRAMAN DASAR

CAPAIAN PEMBELAJARAN MATA KULIAH

- Mahasiswa mampu menyimpulkan topik yang akan dibahas
- Mahasiswa mampu memahami konsep dasar pemrograman
- Mahasiswa mampu memahami tentang algoritma
- Mahasiswa mampu mengenal notasi pada flowchart
- Memahami definisi dan kegunaan variabel, operator, ekspresi, statement
- Memahami perintah dasar input output

SUB CAPAIAN PEMBELAJARAN MATA KULIAH

- Mahasiswa mampu menguasai konsep pemrograman
- Mahasiswa mampu menggunakan notasi flowchart dengan benar
- Melakukan deklarasi variabel
- Menggunakan operator pada ekspresi
- Menggunakan statement
- Menambahkan perintah input dan output pada program

KEBUTUHAN HARDWARE & SOFTWARE

- Computer/Laptop
- Software (Falcon/Dev C++)

MATERI POKOK

❖ FLOWCHART










Flowchart adalah penggambaran secara grafik(bagan & simbol) dari Langkah - langkah prosedur suatu program atau system

Tiap simbol menggambarkan proses, sedangkan antara proses digambarkan garis penghubung.

Kegunaan :

- Membantu melakukan pengecekan bagian yang terlupakan dalam analisis masalah.
- Membantu programmer dalam menganalisis alternatif – alternatif lain dalam pengoperasian.
- Mempermudah penyelesaian suatu masalah khususnya masalah yang perlu dipelajari dan dievaluasi lebih lanjut.
- Membuat rancangan alur suatu program / software.
- Menjadi pedoman bagi programmer dalam Menyusun atau mengembangkan aplikasi.

➤ Symbol – symbol flowchart beserta fungsinya

SIMBOL	NAMA	FUNGSI
	TERMINATOR	Permulaan/akhir program
	GARIS ALIR (FLOW LINE)	Arah aliran program
	PREPARATION	Proses inisialisasi/ pemberian harga awal
	PROSES	Proses perhitungan/ proses pengolahan data
	INPUT/OUTPUT DATA	Proses input/output data, parameter, informasi
	PREDEFINED PROCESS (SUB PROGRAM)	Permulaan sub program/ proses menjalankan sub program
	DECISION	Perbandingan pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya
	ON PAGE CONNECTOR	Penghubung bagian-bagian flowchart yang berada pada satu halaman
	OFF PAGE CONNECTOR	Penghubung bagian-bagian flowchart yang berada pada halaman berbeda

❖ VARIABLE

Variabel adalah suatu tempat yang digunakan untuk menampung data atau konstanta di memori yang mempunyai nilai yang dapat berubah–ubah selama proses program.

sifat dari variabel adalah sementara atau tidak permanen, artinya data atau nilai yang tersimpan dalam variabel akan hilang ketika program dimatikan.

Berikut adalah aturan atau sifat yang dimiliki variabel:

- + Variabel hanya digunakan untuk menyimpan data sementara.
- + Variabel hanya mampu menyimpan satu data atau nilai.

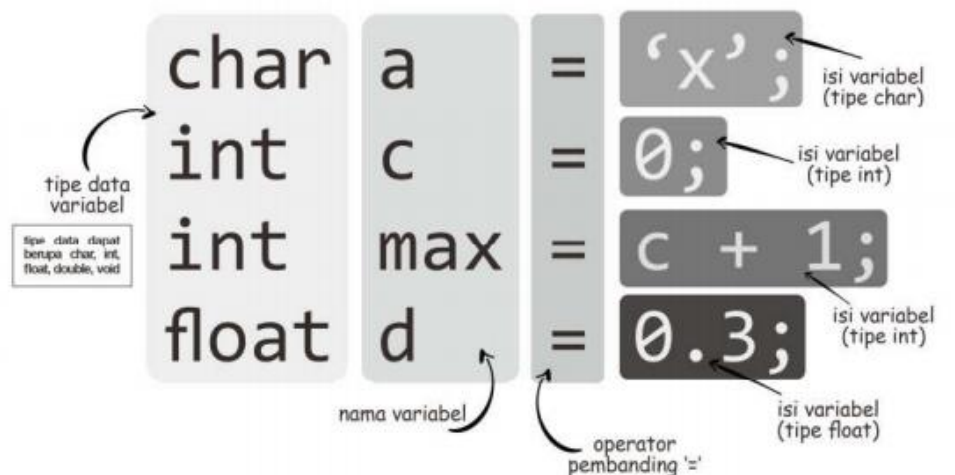
Deklarasi Variabel adalah sebuah statement dengan menyertakan pe data di awal deklarasi variabel. Variabel digunakan agar nilai yang dideklarasikan dapat berubah secara fleksibel. Secara umum tipe- tipe variable ada tiga yaitu:

- Variabel Numerik
- Variabel Text
- Variabel Boolean

Untuk membuat variabel kita harus men-deklarasikan-nya sebelum main program dan sebuah variabel harus ditulis sesuai dengan aturan berikut:

- Karakter pertama harus berupa huruf
- Karakter kedua dan seterusnya dapat berupa angka atau underscore (_)
- Tidak boleh menggunakan spasi

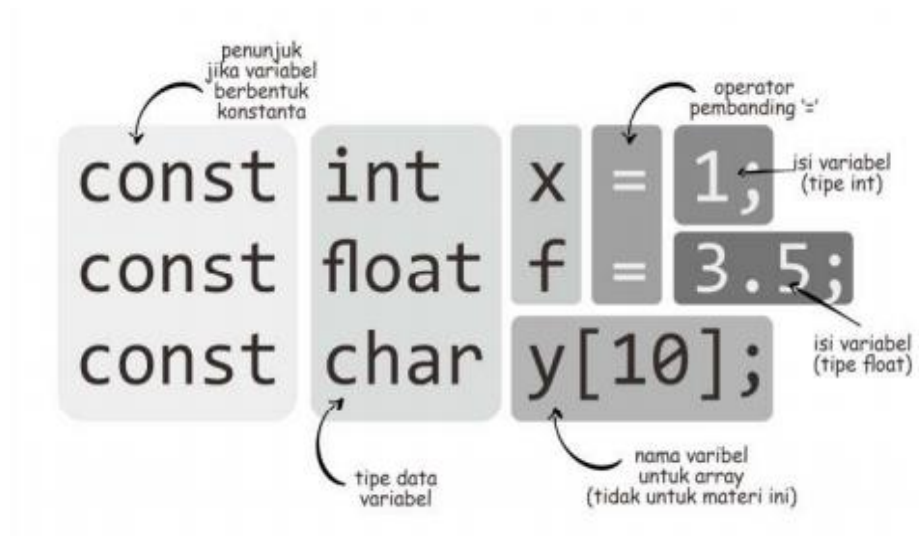
➤ Contoh deklarasi variable :



❖ KONSTANTA

Konstanta adalah sebuah tempat atau container dari suatu nilai. Sesuai dengan namanya, nilai dari konstanta bersifat tetap (konstan) dan tidak bisa diubah sepanjang program berjalan.

➤ Contoh deklarasi konstanta :



❖ TIPE DATA

Tipe data atau kadang disingkat dengan ‘tipe’ saja adalah sebuah pengelompokan data untuk memberitahu *compiler* atau *interpreter* bagaimana programmer ingin mengolah data tersebut.

Macam – macam tipe data dasar terdiri dari :

1. **Char** : adalah tipe data yang berisi 1 huruf atau 1 karakter;
2. **Integer** : adalah tipe data yang berupa angka;
3. **Float** : adalah tipe data yang berupa bilangan pecahan
4. **Double** : adalah tipe data sama seperti **float**, namun double memiliki ukuran penyimpanan yang lebih besar dibandingkan **float**.
5. **Void** : tidak bertipe

No	Tipe Data	Ukuran	Range(Jangkauan)	Format	Keterangan
1	char	1 byte	-128 s/d 127	%c	Karakter/string
2	int	2 byte	-32768 s/d 32767	%d,%i	Integer/bilangan bulat
3	float	4 byte	-3.4E-38 s/d 3.4E+38	%f	Float/bilangan pecahan
4	double	8 byte	-1.7E-308 s/d 1.7+308	%lf	Pecahan presisi ganda
5	void	0 byte	-	-	Tidak bertipe

❖ OPERATOR

Operator adalah simbol yang biasa kita jumpai pada suatu operasi untuk memanipulasi data. Contohnya $a + b$. Dalam operasi tersebut tanda plus (+) berfungsi sebagai operator dalam operasi tersebut. Dalam pemrograman ada banyak jenis operator, yakni :

➤ Operator aritmatik

Simbol Operator	Nama Operator	Contoh	Fungsi
+	Penjumlahan	$\text{Int } z = x + y$	Penjumlahan dari x dan y
-	Pengurangan	$\text{Int } z = x - y$	Pengurangan x oleh y
*	Perkalian	$\text{Int } z = x * y$	Perkalian x dan y
/	Pembagian	$\text{Int } z = x / y$	Pembagian x oleh y
%	Modulus	$\text{Int } z = x \% y$	Sisa dari x dibagi y
++	Increment	$x++$	Sama dengan $x = x + 1$
--	Decrement	$x--$	Sama dengan $x = x - 1$

➤ Operator logika

Simbol Operator	Nama Operator	Contoh	Fungsi
	Atau	(x y)	Operasi dilakukan jika salah satu x atau y bernilai benar atau benar keduanya
&&	Dan	(x && y)	Operasi dilakukan jika kedua nilai x dan y bernilai benar
!	Tidak Benar	(!x)	Operasi dilakukan jika nilai x tidak benar

➤ Operator perbandingan

Simbol Operator	Nama Operator	Contoh	Fungsi
==	Sama dengan	a == b	Benar jika nilai a sama dengan b
!=	Tidak Sama dengan	a != b	Benar jika nilai a tidak sama dengan b
>	Lebih dari	a > b	Benar jika nilai a lebih dari b
<	Kurang dari	a < b	Benar jika nilai a kurang dari b
>=	Lebih dari atau Sama dengan	a >= b	Benar jika nilai a lebih dari atau sama dengan b
<=	Kurang dari atau Sama dengan	a <= b	Benar jika nilai a kurang dari atau sama dengan b

contoh deklarasi operator dan penggunaan tipe data :

```
#include <stdio.h>
#include <stdlib.h>

main()
{
    int i, j;
    float f;

    i = 5; j = 2;
    f = 3.0;

    f = f + j / i;

    printf("value of f is %f\n", f);
    exit(EXIT_SUCCESS);
}
```

```
#include <stdio.h>
#include <stdlib.h>

main()
{
    float i, j;
    float f;

    i = 5; j = 2;
    f = 3.0;

    f = f + j / i;

    printf("value of f is %f\n", f);
    exit(EXIT_SUCCESS);
}
```

output dari program diatas:

```
value of f is 3.000000
Process returned 0   execution time : 0.105 s
Press any key to continue.
```

Contoh 1

```
value of f is 3.400000
Process returned 0   execution time : 0.124 s
Press any key to continue.
```

Contoh 2

Pembahasan : Walaupun operasi dan nilai dari semua variable kedua contoh program sama, namun output yang dihasilkan berbeda. Hal ini dikarenakan tipe data variabel i dan j pada Contoh 1 adalah integer sehingga hasil operasi dari j / i sama dengan 0. Berbeda dengan Contoh 2 dengan tipe data variable i dan j adalah float maka hasil dari j / i sama dengan 0,4.

❖ STANDAR I/O

Input & Output (scanf & prin) adalah akfitas pengguna dengan program pada komputer yang memungkinkan kita memberikan data kedalam program dan program itu juga dapat mengeluarkan data untuk pengguna.

Syntax	Fungsi	Penggunaan
printf();	Menampilkan data pada output dengan format yang ditentukan (lihat pada tipe data)	<code>printf(format_tipe_data, variabel);</code> <code>printf("kalimat");</code>
scanf();	Membaca data dari output dengan format yang ditentukan (lihat pada tipe data)	<code>scanf(format_tipe_data, variabel_penyimpanan);</code>

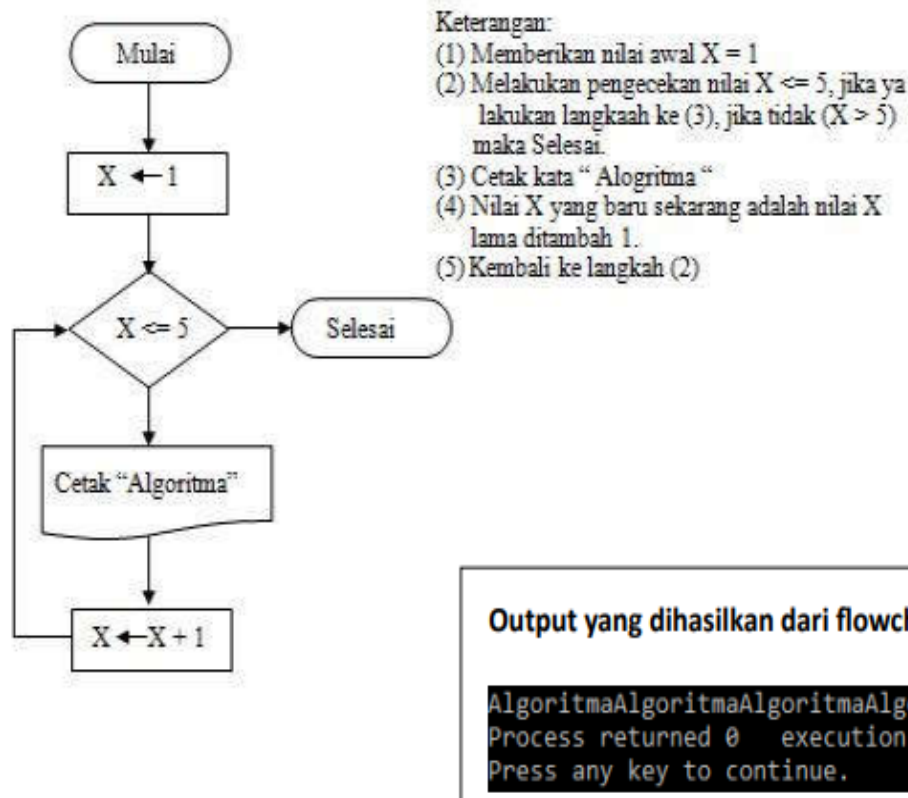
Contoh deklarasi I/O :

```

1  #include <stdio.h>
2
3  main () {
4      int a,b;
5
6      printf("masukkan nilai a : ");
7      scanf("%d", &a);
8      printf("masukkan nilai b : ");
9      scanf("%d", &b);
10
11     printf("nilai a = %d\n", a);
12     printf("nilai b = %d\n", b);
13
14     return 0;
15 }
```

MATERI PRAKTIKUM

Contoh flowchart :



Untuk membuat konstanta dalam bahasa C, terdapat 2 cara. Yakni menggunakan keyword **#define** atau menggunakan keyword **const**. penggunaan **#define** sendiri ditujukan untuk mempersingkat waktu compile karena define merupakan preprocessor.

Contoh :

```

1  #include <stdio.h>
2  #define panjang 8
3  #define lebar 5
4
5  int main (void){
6      printf("panjang = %d", panjang);
7      printf("\n");
8      printf("lebar = %d", lebar);
9      printf("\n");
10     printf("luas ( panjang*lebar ) = %d", panjang*lebar);
11     printf("\n");
12
13     return 0;
14 }
  
```


PRE-PRAKTIKUM

```

1  #include <stdio.h>
2  #define a 2.5
3
4  int main()
5  {
6      int nim,bil1,bil2,bil3,
7      jumlah;
8      float hasil;
9
10     printf( 'masukkan 3 digit terakhir nim anda : ');
11     scanf ("%d", &nim);
12     printf( "masukkan bilangan pertama : ");
13     scanf ("%d", bil1);
14     printf( "masukkan bilangan kedua : ");
15     scanf ("%f", &bil2);
16     jumlah = nim -(bil1+bil2)
17
18     printf("jumlah yang diperoleh : %d \n", jumlah);
19     printf("\nmasukkan bilangan ketiga : ");
20     scanf ("%d", &bil3);
21     hasil = jumlah*bil3\a
22     printf( "\nnim anda adalah : %d /n", nim );
23     printf("hasil yang anda peroleh dari data diatas adalah : %f /n", hasil);
24
25
26     return 0;
27 }

```

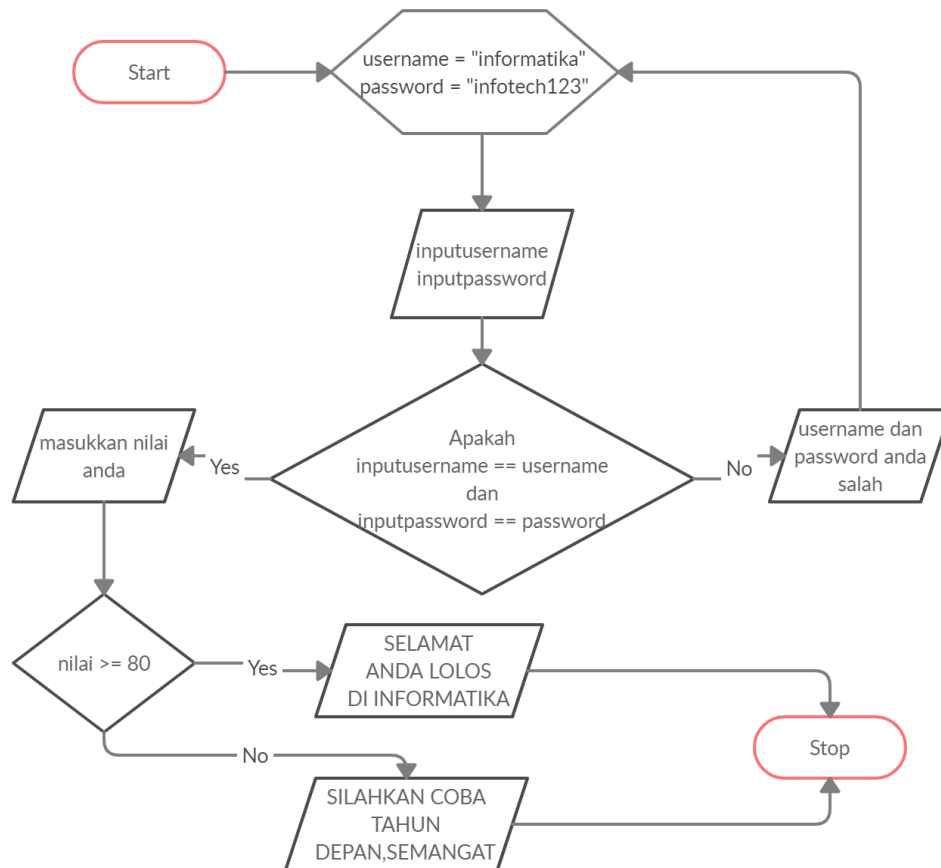
Terdapat beberapa kesalahan dalam baris-baris program diatas, temukan kesalahannya pada baris ke berapa setelah itu jelaskan mengapa penulisan baris program tersebut salah kemudian perbaiki!

Salin baris program yang mengalami kesalahan penulisan ke dalam bentuk word kemudian tulis dibawahnya penjelasan mengapa baris tersebut salah. Simpan juga program yang sebelum dan sesudah diperbaiki.

Simpan semua berkas Word maupun program dalam satu folder yang di extract dalam bentuk Zip kemudian kumpulkan ke dalam attachment I-Lab.

LEMBAR KERJA

KEGIATAN 1



Jelaskan alur flowchart diatas kepada asisten !

KEGIATAN 2

```

>>> NILAI PRAKTIKUM <<<

masukkan nilai 1 : 95
masukkan nilai 2 : 87
masukkan nilai 3 : 90
nilai akhir 4 persen dari rata rata nilai yang diperoleh

rata-rata nilai yang kamu peroleh adalah : 90
nilai akhir yang kamu peroleh adalah : 3.60

Process returned 0   execution time : 11.746 s
Press any key to continue.

```

Gambarkan flowchart dan Buatlah output program seperti di atas dengan mengimplementasikan operator dan constant

*nilai 1 -3 merupakan inputan dari user dan untuk nilai akhir sebesar 4 persen bertipe konstanta

RUBRIK PENILAIAN

Pre-praktikum

Kriteria	Poin
Soal Pre-Praktikum Dikerjakan Dengan Baik	10
Soal Pre-Praktikum Tidak Dikerjakan	-5

Kegiatan 1

Kriteria	Poin
Gambar simbol benar (input, output, proses, kondisi, looping, start-end, anak panah)	20
Ketepatan menjawab pertanyaan	10
Total	30

Kegiatan 2

Kriteria	Poin
Berhasil di compile dan running tanpa error	10
Pemilihan tipe data sesuai dengan nilai yang ingin disimpan	10
Sintaks deklarasi konstanta sudah benar	15
Penggunaan operator aritmatika (+, -, *, /, %)	10
Ketepatan penggambaran flowchart	15
Total	60