

VARIABLE & TIPE DATA

Informatika

DISAJIKAN OLEH

Johannes Alexander
Putra

DAFTAR ISI

Capaian Pembelajaran

01

Tujuan Pembelajaran

02

Variable

03

Tipe Data

08

Input dan Output

11

Daftar Pustaka

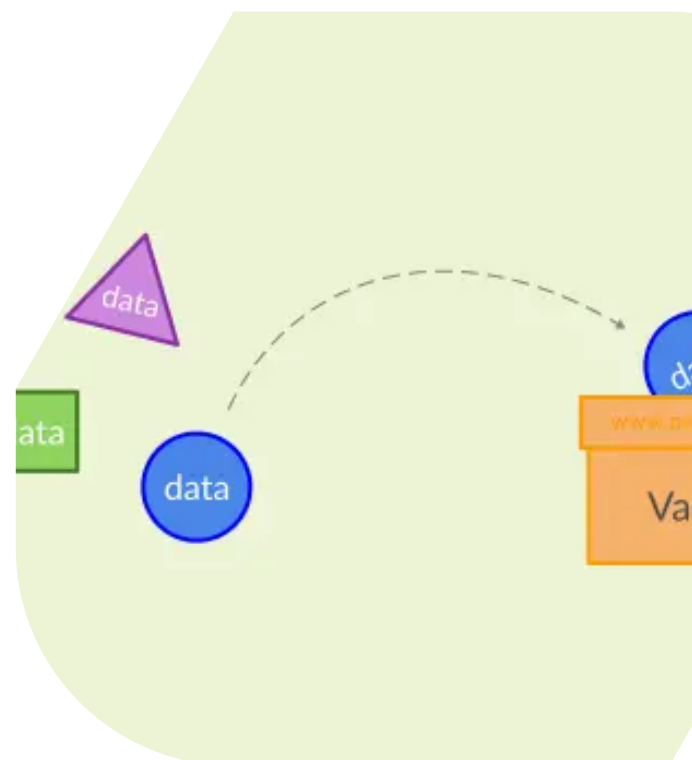
15

CAPAIAN PEMBELAJARAN

Pada akhir fase E, peserta didik mampu menerapkan praktik baik konsep pemrograman prosedural dalam salah satu bahasa pemrograman prosedural dan mampu mengembangkan program yang terstruktur dalam notasi algoritma atau notasi lain, berdasarkan strategi algoritmik yang tepat.

TUJUAN PEMBELAJARAN

- Memahami permasalahan yang berkaitan dengan variable
- Menentukan pemecahan masalah dengan variable
- Melakukan implementasi variable untuk suatu permasalahan
- Melakukan evaluasi terhadap penggunaan variable
- Memahami permasalahan yang berkaitan dengan tipe data
- Menentukan pemecahan masalah dengan tipe data
- Melakukan implementasi flowchart untuk suatu tipe data
- Melakukan evaluasi terhadap penggunaan tipe data
- Memahami permasalahan yang berkaitan dengan input output
- Menentukan pemecahan masalah dengan input output
- Melakukan implementasi flowchart untuk suatu input output
- Melakukan evaluasi terhadap penggunaan input output



VARIABLE

Variable menurut Jubilee Enterprise (2017) adalah salah satu bagian dalam bahasa pemrograman yang paling mendasar. Dengan menggunakan variable pemrogram dapat menyimpan data ke tempat sementara dan mengolah data tersebut menggunakan variable.

Variable menurut Sukamto (2016) adalah tempat untuk menyimpan data dengan tipe tertentu yang isinya dapat diubah-ubah sesuai dengan tipenya. Setiap variable hanya dapat menyimpan satu buah nilai. Sehingga jika nilai itu diubah maka nilai sebelumnya diganti dengan yang baru. Sedangkan ada juga yang dikenal dengan konstanta.

Konstanta adalah variable yang ditentukan nilai standarnya dari awal dan biasanya nilainya tidak diubah-ubah. Dalam logika kita variable dapat digambarkan dengan sebuah kotak kosong yang dapat diisi dengan sesuatu dengan jenis tertentu.

DEKLARASI & INISIALISASI

Deklarasi merupakan bagian yang digunakan untuk mengumumkan semua nama yang dipakai dalam algoritma beserta propertynya. Nama yang dideklarasikan adalah:

- Nama Konstanta
- Nama Peubah
- Nama tipe bentukan
- Nama prosedur/fungsi

Munir, R. & Lidya, L (2016)

Contoh dalam pseudocode:

DEKLARASI

m : **integer**

Contoh dalam C++

```
int m
```

Inisialisasi adalah tahapan pengisian nilai awal suatu objek pemrograman.

Contoh dalam pseudocode

m <- 10

Contoh dalam C++

```
m = 10
```

KELEMAHAN TANPA VARIABLE

- Kode tidak fleksibel
- Untuk mengganti nama "Alex," Anda harus melakukan banyak perubahan di banyak tempat :
 - Pengeditan yang melelahkan
 - Risiko kehilangan "Alex"

```
cout<<"My name is Alex\n";  
cout<<"Alex is so cool!\n";  
cout<<"Hooray Alex!\n";
```

Gambar 1. Contoh Output tanpa variable pada C++

KEUNTUNGAN VARIABLE

- Kode menjadi fleksibel
 - Mengingat dan memanipulasi nilai
- Untuk mengganti nama "Alex," Anda buat satu perubahan:
 - Pengeditan yang efisien
 - Tidak ada risiko kehilangan "Alex"

```
string x = "Sam";  
cout<<"My name is " + x + "\n";  
cout<<x + " is so cool!";
```

Gambar 2. Contoh Output dengan variable pada c++

Contoh Soal

Anton merupakan seorang peternak. Ia memiliki 3 hewan ternak yaitu ayam, bebek, dan sapi. Anton memerlukan wadah untuk menampung jumlah dari telur ayam, telur bebek, dan jumlah dari susu sapi

Step 1- Memahami Masalah

Memahami masalah: Menentukan apa yang diketahui dan ditanyakan kemudian memberikan keterangan apa yang sudah diketahui cukup untuk menjawab pertanyaan.

Soal tersebut adalah soal untuk deklarasi variable untuk kasus 3 hewan ternak.
Keterangan pada soal sudah cukup untuk menjawab soal

Step 2- Mengidentifikasi Masalah

Merencanakan penyelesaian masalah : Mengidentifikasi masalah dan mencari jalan yang tepat untuk menyelesaikannya

Masalah dalam soal tersebut kurang lebih prinsipnya sama seperti menyediakan 3 wadah berbeda

Step 3- Melaksanakan Pemecahan Masalah

Melaksanakan pemecahan masalah: menekankan pelaksanaan penyelesaian masalah dengan memeriksa apakah setiap langkah sudah tepat atau belum

```
#include <iostream>
|
using namespace std;

int main()
{
    int jumlahTelurAyam;
    int jumlahTelurBebek;
    int jumlahSusuSapi;
}
```


Step 4-Memeriksa Kembali

Memeriksa kembali hasil, langkah ini dilakukan dengan memeriksa kebenaran jawaban

Pada bagian ini kita harus memeriksa kembali

1. Harus ada 3 variable karena menampung 3 hal yang berbeda
2. Berupa bilangan bulat maka digunakan int

TIPE DATA

Tipe data sudah dikenal dalam kehidupan sehari-hari. Tipe data ini sudah ada sejak zaman dahulu. Setiap hari kita berbicara mengenai angka dan karakter. Contoh tipe data adalah bilangan bulat, bilangan riil, bilangan logika, karakter, string, dll (Sukanto, 2018).

BILANGAN BULAT

tipe bilangan bulat biasanya disebut sebagai integer akan tetapi pada aplikasinya tipe ini tidak hanya dinyatakan sebagai integer saja. Ada tipe data lain seperti long dan short.

Tabel 1 Tipe Data Bilangan Bulat

Tipe Data Bilangan Bulat	Jangkauan pada Bahasa C++
Long	-2147483648,..., -2147483647
Integer/int	-32768,...,+32767
short	-128,..., +127

Bilangan bulat juga dapat menampung bilangan ASCII

1		17		33	!	49	1	65	A	81	Q	97	a	113	q
2		18		34	"	50	2	66	B	82	R	98	b	114	r
3		19		35	#	51	3	67	C	83	S	99	c	115	s
4		20		36	\$	52	4	68	D	84	T	100	d	116	t
5		21		37	%	53	5	69	E	85	U	101	e	117	u
6		22		38	&	54	6	70	F	86	V	102	f	118	v
7		23		39	'	55	7	71	G	87	W	103	g	119	w
8		24		40	(56	8	72	H	88	X	104	h	120	x
9		25		41)	57	9	73	I	89	Y	105	i	121	y
10		26		42	*	58	:	74	J	90	Z	106	j	122	z
11		27		43	+	59	;	75	K	91	[107	k	123	{
12		28		44	,	60	<	76	L	92	\	108	l	124	
13		29		45	-	61	=	77	M	93]	109	m	125	}
14		30		46	.	62	>	78	N	94	^	110	n	126	~
15		31		47	/	63	?	79	O	95	_	111	o	127	
16		32		48	0	64	@	80	P	96	`	112	p	128	€

Misalnya :

```
int a = 'A'
```

maka nilai a sama dengan 65

BILANGAN RIIL

bilangan riil biasanya digunakan untuk menyatakan bulangan yang membutuhkan ketelitian atau ada angka dibelakang koma (Sukamto,2018) . Pada bahasa pemrograman java biaanya menggunakan double. Akan tetapi dalam pemrograman C++ biasanya menggunakan keduanya baik float maupun double

Tabel 2 float

Bahasa Algoritmik	Bahasa C++
kurs: riil	float kurs;
kurs <-1.02	kurs = 1.02;

BOOLEAN

Boolean adalah tipe data untuk menyatakan pernyataan benar atau salah. Tipe data ini hanya dapat diisi dengan dua buah nilai bisa true atau bisa false (Sukamto, 2018).

Tabel 3 Boolean

Bahasa Algoritmik	Bahasa C++ & Java
kupas: boolean	bool kupas;
kupas <-true	kupas = true;

CHAR

Karakter atau char pada bahasa pemrograman merupakan tipe data yang menyimpan **sebuah karakter**. Karakter ditandai dengan kutip tanda satu " (Sukamto, 2018)

Tabel 4 Char

Bahasa Algoritmik	Bahasa C++ & Java
kupas: char	char kupas;
kupas <-‘Y’	kupas = ‘Y’;

STRING

String adalah tipe data yang merupakan kumpulan dari karakter. Dalam bahasa Java dan C++ ditulis dengan kutip dua ""

Tabel 5 String

Bahasa Algoritmik	Bahasa C++
kupas: string	string kupas;
kupas <- "ya"	kupas = "ya";

String pada bahasa pemrograman java adalah objek dan bukan tipe data primitif. Oleh karena itu ditulis dengan huruf besar saat mendeklarasikan tipe data String.

Contoh:

String kupas

PERLU DIINGAT

Pendeklarasian variable adalah sangat case sensitive (huruf besar dan huruf kecil sangat berpengaruh). ingat seluruh tipe data data yang sebelumnya dijelaskan ditulis dalam huruf kecil.

penulisan: String x = "Hello" dalam bahasa pemrograman C++ adalah salah karena String seharusnya ditulis string. Kemudian setiap baris kode akan dipisahkan dengan semicolon (;) yang menandakan baris tersebut sudah selesai.

Penulisan string x dan string X juga berbeda karena ingat C++ adala bahasa yang sangat case sensitive sehingga string x dan string X dianggap variable yang berbeda

INPUT DAN OUTPUT

Nilai dari sebuah variable dapat diisi dengan masukan dari keyboard menggunakan fungsi yang sudah ada pada pustaka (Library) (Sukamto, 2018). C++ menggunakan

```
#include <iostream>
using namespace std;
```

Tabel 6 input pada C++ dan Java

Bahasa Algoritmik	Bahasa C++
kupas: string	string kupas;
input(kupas)	cin>>kupas;

Sebuah program komputer dapat menampilkan kalimat ke layar. Hal ini biasanya dilakukan untuk menampilkan perintah masukan.

Tabel 7 output pada C++ dan Java

Bahasa Algoritmik	Bahasa C++
kupas: string	string kupas;
output("Masukan isi variable kupas");	cout<<"Masukan isi variable kupas"

INPUT DAN OUTPUT

Simak perbedaan output berikut ini:

```
#include <iostream>

using namespace std;

int main()
{
    cout<<"Hello World" <<endl;
    cout<<"Hello World";
}
```

```
#include <iostream>

using namespace std;

int main()
{
    cout<<"Hello World";
    cout<<"Hello World";
}
```

Hasilnya:

```
Hello World
Hello World
```

```
Hello WorldHello World
```

Dengan menggunakan endl kita menambahkan baris baru.

Hal ini juga bisa dengan cout<<"Hello World \n"; ataupun Cout <<"Hello World"<<'\\n';

```
8 *****
9 #include <iostream>
10
11 using namespace std;
12
13 int main()
14 {
15     cout<<"Hello World \n";
16     cout<<"Hello World \n";
17
18
19 }
```

```
ello World
ello World
```

```
9 #include <iostream>
10
11 using namespace std;
12
13 int main()
14 {
15     cout<<"Hello World"<<'\\n';
16     cout<<"Hello World"<<'\\n';
17
18
19 }
```

```
Hello World
Hello World
```

CONTOH SOAL

Sebuah program komputer diminta untuk dapat memasukan dan mengeluarkan sebuah kalimat dan nama variabelnya kalimat

Step 1- Memahami Masalah

Memahami masalah: Menentukan apa yang diketahui dan ditanyakan kemudian memberikan keterangan apa yang sudah diketahui cukup untuk menjawab pertanyaan.

Soal tersebut adalah soal untuk mengambil input dari keyboard dan mengeluarkannya.
Keterangan pada soal sudah cukup untuk menjawab soal

Step 2- Mengidentifikasi Masalah

Merencanakan penyelesaian masalah : Mengidentifikasi masalah dan mencari jalan yang tepat untuk menyelesaikannya

Masalah dalam soal tersebut kurang lebih prinsipnya sama seperti membuat suatu variable dan nanti diisi data lewat keyboard dan ditampilkan ke layar datanya

Step 3- Melaksanakan Pemecahan Masalah

Melaksanakan pemecahan masalah: menekankan pelaksanaan penyelesaian masalah dengan memeriksa apakah setiap langkah sudah tepat atau belum

```
#include <iostream>

using namespace std;

int main()
{
    string kalimat;
    cin>>kalimat;
    cout<<kalimat;
    return 0;
}
```

Step 4-Memeriksa Kembali

Memeriksa kembali hasil, langkah ini dilakukan dengan memeriksa kebenaran jawaban

Pada bagian ini kita harus memeriksa kembali

1. Pada program tersebut penamaan tipe datanya sudah tepat yaitu string pada C++.
2. Pada C++ sudah mengimport library yang tepat
3. Input dan outputnya sudah tepat

DAFTAR PUSTAKA

- Polya, G. (2015). How to Solve It: A New Aspect of Mathematical Method. New Jersey: Princeton University Press.
- Sukamto, R. A. (2018). Logika Algoritma dan Pemograman Dasar. Bandung: Modula.
- Liang. (2015). Introduction to Java Programming Comprehensive Version 10 Edition. Pearson
- Wahyono, dkk. (2021). Buku Panduan Guru Informatika SMA Kelas 10. Jakarta: Pusat Kurikulum dan Perbukuan
- Jubilee Enterprise. Belajar VB, Visual C#, dan Python Menggunakan Visual Studio. (2017). Jakarta: PT Elex Media Komputindo
- Munir, R. & Lidya, L. (2016). Algoritma dan Pemrograman Dalam Bahasa Pascal, C, dan C++. Edisi 6. Bandung : Informatika