

# Variables, Data Type and Operators II, flowchart

# Variable

- A **place to store** an item of data that can vary in a program,
- Every variable has a name, it's called **Variable Name**,
- Use that **name** to refer to that place in memory to retrieve what it contains or store a new data value there,

# Variables Declaration And Assigning

- `DATA_TYPE variable_name;`
- `Assigning = (assignment)`
- Eq.

```
int a = 1;
```

```
float f = 10.0;
```

```
double d = 20.0;
```

# Role (Variable Name)

- Can't begin with a digit,
- Can't include any other characters besides letters, underscores, and digits,
- Variable names are case sensitive
- Cannot use **keywords**

# Data Type

Type	Typical Size in Bits	Minimal Range
char	8	-127 to 127
unsigned char	8	0 to 255
signed char	8	-127 to 127
int	16 or 32	-32,767 to 32,767
unsigned int	16 or 32	0 to 65,535
signed int	16 or 32	Same as <b>int</b>
short int	16	-32,767 to 32,767
unsigned short int	16	0 to 65,535
signed short int	16	Same as <b>short int</b>
long int	32	-2,147,483,647 to 2,147,483,647
long long int	64	$-(2^{63} - 1)$ to $2^{63} - 1$ (Added by C99)
signed long int	32	Same as <b>long int</b>
unsigned long int	32	0 to 4,294,967,295
unsigned long long int	64	$2^{64} - 1$ (Added by C99)
float	32	1E-37 to 1E+37 with six digits of precision
double	64	1E-37 to 1E+37 with ten digits of precision
long double	80	1E-37 to 1E+37 with ten digits of precision

**Table 2 -1.** All Data Types Defined by the C Standard

# Conversion Format To Print Variable

- %d : int,long,short,etc.
- %f : float,double,long double.
- %c : char.

# Sample

```
#include <stdio.h>
#include <conio.h>
main()
{
    int a = 1;
    printf("Nilai a adalah %d ",a);
}
```

# Casting

- The ability to explicitly convert a value of one type to another type.
- Sample

```
int a = 1;
```

```
int b = 2;
```

```
float c;
```

```
c = a/b;
```

```
printf("%f", c);
```

**Output : 0.000000**



## Casting (cont)

```
int a = 1;  
int b = 2;  
float c;  
c = (float) a/b;  
printf("%f", c);
```

Output : 0.500000

# Operators

- Arithmetic
- Logical
  - Relational
  - Equality

# Arithmetic Operator

**Table 2-1.** *Basic Arithmetic Operators*

<b>Operator</b>	<b>Action</b>
+	Addition
-	Subtraction
*	Multiplication
/	Division
%	Modulus

# Algebra (Al Jabar) to C

- Algebra =  $a + b$  , C = a + b
- Algebra =  $a - b$  , C = a - b
- Algebra =  $ab$  , C = a \* b
- Algebra =  $\frac{a}{b}$  , C = a/b
- $Y = \frac{ax^2 + bx + c}{\sqrt{9}}$  , C = (a\*x\*x) + (b\*x) + c
- , C = ?

# Library : math.h

- Algebra =  $\sqrt{x}$ , C = sqrt(x)

```
#include <stdio.h>
#include <conio.h>
#include <math.h>
main()
{
    float a = 9.00;
    float akar_a = sqrt(a);
    printf("akar dari %f adalah %f
    ", a, akar_a);
    getch();
}
```

# Math.h

- Algebra =  $x^y$  , C = pow(x,y)
- Algebra =  $\sin x$  , C = sin(x)
- Algebra =  $e^x$  , C = exp(x)

## Reference

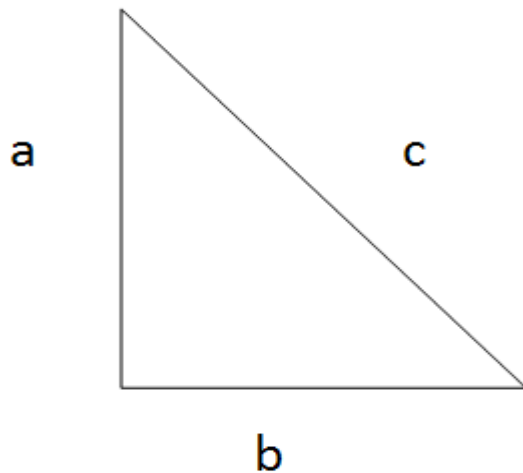
- C++ How To Program
- Beginning C From novice to professional
- C The Complete Reference

# Excercise

- Algebra  $= \frac{\sqrt{b^2 - 4ac}}{2a}$  , C = ?

- Algebra  $= \frac{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}}{1}$  , C = ?

- Algebra  $= \frac{1}{(1 - \sin \alpha)}$  , C = ?



- Algebra : C = ?

Question ?



# Logical Operator

Standard algebraic equality or relational operator	C++ equality or relational operator	Sample C++ condition	Meaning of C++ condition
<i>Relational operators</i>			
>	>	<code>x &gt; y</code>	x is greater than y
<	<	<code>x &lt; y</code>	x is less than y
≥	>=	<code>x &gt;= y</code>	x is greater than or equal to y
≤	<=	<code>x &lt;= y</code>	x is less than or equal to y
<i>Equality operators</i>			
=	==	<code>x == y</code>	x is equal to y
≠	!=	<code>x != y</code>	x is not equal to y

# Relational

- Statement  $1 < 2$  result : **1**
- Statement  $2 < 1$  result : **0**
- Statement  $1 == 1$  result : ...
- Statement  $1 != 2$  result : ...
- Statement  $1 <= 2$  result : ...
- Statement  $10 <= 10$  result : ...

# Logical Operator && (AND)

a	b	a && b	result
0	0	0 && 0	0
0	1	0 && 1	0
1	0	1 && 0	0
1	1	1 && 1	1

# Logical Operator || (OR)

a	b	a    b	result
0	0	0    0	0
0	1	0    1	1
1	0	1    0	1
1	1	1    1	1

# Operator Precedence

1. Relational

2. Logika

i. &&

ii. ||

• Sample (true/false) :

1 >= 0 || 2 <=3 && 5==5

# Answer

1 >= 0 || 2 <=3 && 5==5

true || 2 <=3 && 5==5

true || true && 5==5

true || true && true

true || true

**true**

# Logical Operator ! (NOT)

- $!1 = 0$
- $!0 = 1$
- $!true = false$
- $!false = true$

# Sample

- `(1 <= 10) , result = true`
- `(1 <= 10) || (10 != 1) , result = ?`
- `(!true) || (10 >= 1) , result = ?`
- `(false) || (10 >= 1) , result = ?`
- `(true) && (false) , result = ?`
- `(true) && (true) , result = ?`
- `((!true && !false) || (1 < 0))  
&& ((!false || (true))) , result =  
?`





Question ?

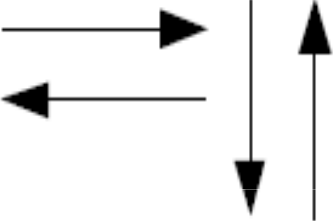
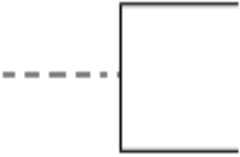
# Flowchart

- Representasi grafis dari langkah – langkah yang harus diikuti dalam menyelesaikan suatu permasalahan yang terdiri atas sekumpulan simbol, dimana masing – masing simbol merepresentasikan kegiatan tertentu,
- Flowchart diawali dengan penerimaan input dan diakhiri dengan penampilan output ,
- Sebuah flowchart pada umumnya tidak menampilkan instruksi bahasa pemrograman, namun menetapkan konsep solusi dalam

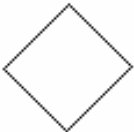

# Simbol Flowchart

<b>Simbol</b>	<b>Nama</b>	<b>Pengertian</b>
	<b>Simbol Proses</b>	Simbol ini digunakan untuk melambangkan kegiatan pemrosesan input. Dalam simbol ini, kita dapat menuliskan operasi-operasi yang dikenakan pada input, maupun operasi lainnya. Sama seperti aturan pada simbol input, penulisan dapat dilakukan secara satu per satu maupun secara keseluruhan.
	<b>Simbol Input - Output (IO)</b>	Merepresentasikan fungsi I/O yang membuat sebuah data dapat diproses (input) atau ditampilkan (output) setelah mengalami eksekusi informasi



# Simbol Flowchart (lanjutan)

	<b>Simbol Garis Alir</b>	Simbol ini digunakan untuk menghubungkan setiap langkah dalam flowchart dan menunjukkan kemana arah aliran diagram. Anak panah ini harus mempunyai arah dari kiri ke kanan atau dari atas ke bawah. Anak panah ini juga dapat diberi label, khususnya jika keluar dari simbol percabangan.
	<b>Simbol Anotasi</b>	Merepresentasikan informasi deskriptif tambahan, komentar atau catatan penjelasan. Dalam simbol ini, kita dapat menuliskan komentar apapun dan sebanyak apapun, hal ini berguna untuk memperjelas langkah-langkah dalam flowchart. Garis vertical dan garis terputus - putus dapat ditempatkan pada sisi kanan maupun kiri.


# Simbol Flowchart (lanjutan)

	<b>Simbol Percabangan</b>	Simbol ini digunakan untuk melambangkan percabangan, yaitu pemeriksaan terhadap suatu kondisi. Dalam simbol ini, kita menuliskan keadaan yang harus dipenuhi. Hasil dari pemeriksaan dalam simbol ini adalah YES atau NO. Jika pemeriksaan menghasilkan keadaan benar, maka jalur yang harus dipilih adalah jalur yang berlabel Yes, sedangkan jika pemeriksaan menghasilkan keadaan salah, maka jalur yang harus dipilih adalah jalur yang berlabel No.
	<b>Simbol Terminator</b>	Terminator berfungsi untuk menandai awal dan akhir dari suatu flowchart. Simbol ini biasanya diberi label START untuk menandai awal dari flowchart, dan label STOP untuk menandai akhir dari flowchart. Jadi dalam sebuah flowchart pasti terdapat sepasang terminator yaitu terminator start dan stop.

# Simbol Flowchart (lanjutan)

	<b>Simbol Konektor</b>	<p>Simbol konektor digunakan pada waktu menghubungkan suatu langkah dengan langkah lain dalam sebuah flowchart dengan keadaan on page atau off page. On page connector digunakan untuk menghubungkan suatu langkah dengan langkah lain dari flowchart dalam satu halaman, sedangkan off page connector digunakan untuk menghubungkan suatu langkah dengan langkah lain dari flowchart dalam halaman yang berbeda. Connector ini biasanya dipakai saat media yang kita gunakan untuk menggambar flowchart tidak cukup luas untuk memuat gambar secara utuh, jadi perlu dipisah-pisahkan. Dalam sepasang connector biasanya diberi label tertentu yang sama agar lebih mudah diketahui pasangannya.</p>
	<b>Simbol Prosedur</b>	<p>Simbol ini berperan sebagai blok pembangun dari suatu program. Prosedur memiliki suatu flowchart yang berdiri sendiri diluar flowchart utama. Jadi dalam simbol ini, kita cukup menuliskan nama prosedurnya saja, jadi sama seperti jika kita melakukan pemanggilan suatu prosedur pada program utama (main program). Sama dengan aturan pada simbol percabangan, penulisan nama prosedur dilakukan secara satu per satu.</p>

# Simbol Flowchart (lanjutan)

Pemberian nilai awal suatu variabel <i>(Preparation)</i>	
--	---

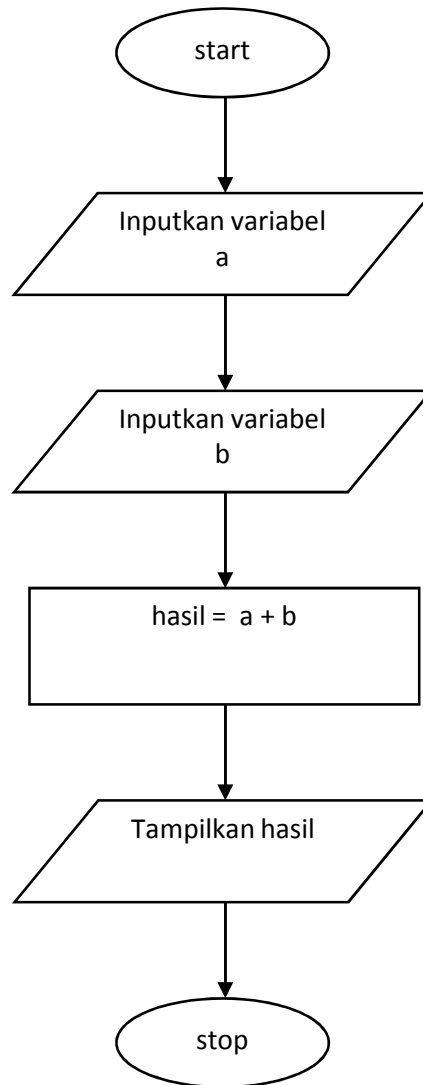


# Sample

- Buatlah flowchart dari program penambahan dua buah bilangan. Ctt. Kedua variabel dimasukkan melalui keyboard.



# Jawab



# Latihan

- Buatlah flowchart untuk program perhitungan luas segitiga, dimana nilai alas dan tinggi telah ditentukan.
- Buatlah flowchart untuk program perhitungan luas lingkaran, dimana jari-jari dimasukkan melalui keyboard.