



Programming with C and C++

CSC-101 (Lecture 03)

Dr. R. Balasubramanian
Professor

Department of Computer Science and Engineering
Mehta Family School of Data Science and Artificial Intelligence
Indian Institute of Technology Roorkee
Roorkee 247 667

bala@cs.iitr.ac.in

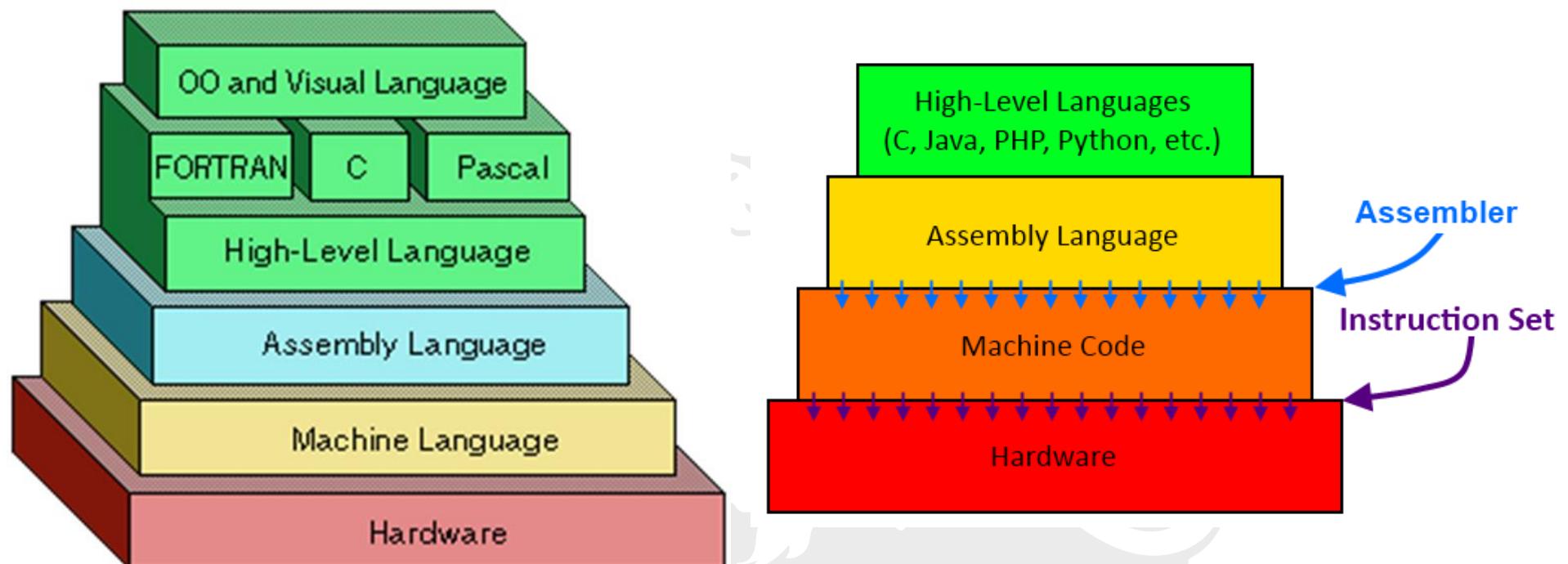
<https://faculty.iitr.ac.in/cs/bala/>



Low-level/High-level languages



175 YEARS OF
IIT ROORKEE
Estd. 1847



Low-level/High-level languages



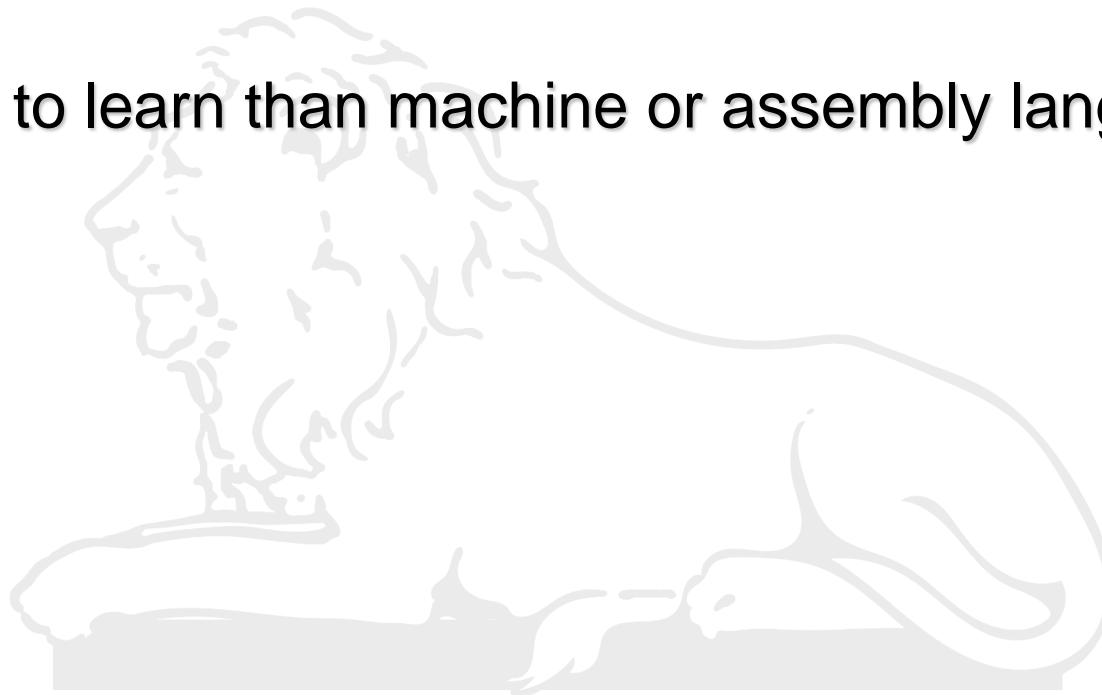
- ▶ Machine language and Assembly language are both
 - Microprocessor specific (**Machine dependent**)
so they are called
 - Low-level languages

- ▶ **Machine independent** languages are called
 - High-level languages
 - For e.g. BASIC, PASCAL, C++, C, JAVA, etc.
 - A software called **Compiler** is required to convert a high-level language program to machine code
 - **Compiler** is a software system that translates program into machine language so that Computer Operating System (OS) can then run.

High-Level languages



- Instructions are quite English-like, and a single instruction can be written to correspond to many operations at the machine level.
- Easier to learn than machine or assembly languages.



Examples



- ❖ **COBOL** – developed in the 1960s for business transactions.
- ❖ **FORTRAN** – developed for mathematical calculations.
- ❖ **Pascal** - is a structured high-level language.
- ❖ **C** – is designed to support only procedure-oriented programming. Popular language for developing system applications such as **operating system** and **compilers**.
- ❖ **C++** - is extension of C programming language that support object oriented programming and procedure-oriented approach.
- ❖ **Java** – is an object-oriented language.

Algorithm

- A set of **explicit, unambiguous finite steps**, which when carried out for a given set of **initial condition** to produce the corresponding **output** and terminate in **finite time**.

- ❖ Write an algorithm to find the sum of 3 numbers.

- 1) **Input A, B, C**
- 2) **Sum = A + B + C**
- 3) **Print Sum**



❖ Write an algorithm to find the sum of 3 numbers.

- 1) Input A, B, C
- 2) Sum = A + B + C
- 3) Print Sum

{ Algo' . }

Algorithm



- Ex. 2: Write an algorithm to swap two numbers

Step 1: Start

step 2: read 'a' and 'b' values

step 3: interchange the values

temp=a

a=b

b=temp

step 4: write a and b values

step 5: stop

Algorithm



- Ex. 3: Write an algorithm to swap two numbers without using temporary variable

Step 1: Start

step 2: read 'a' and 'b' values

step 3: interchange the values

$$a=a+b$$

$$b=a-b$$

$$a=a-b$$

step 4: write a and b values

step 5: stop

Flowcharts



- ▶ Flowchart is a pictorial/graphical representation of an algorithm
- ▶ Uses symbols (boxes of different shapes) that have standardized meanings to denote different types of instructions
- ▶ Actual instructions are written inside boxes
- ▶ Boxes are connected by solid lines having arrow marks to indicate the exact sequence in which the instructions are to be executed

Basic Flowcharts Symbols



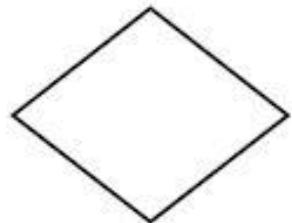
Terminal



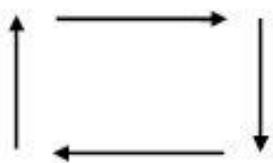
Input/Output



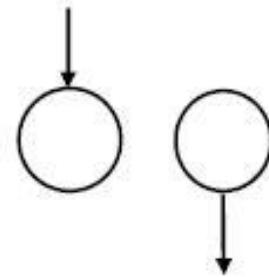
Processing



Decision

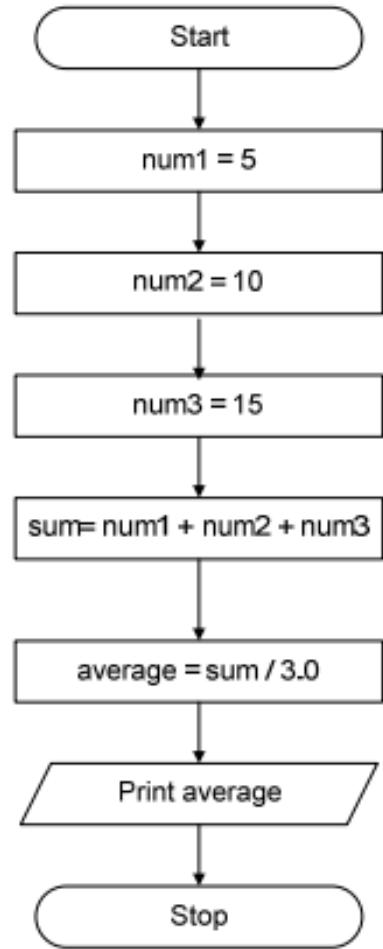


Flow lines



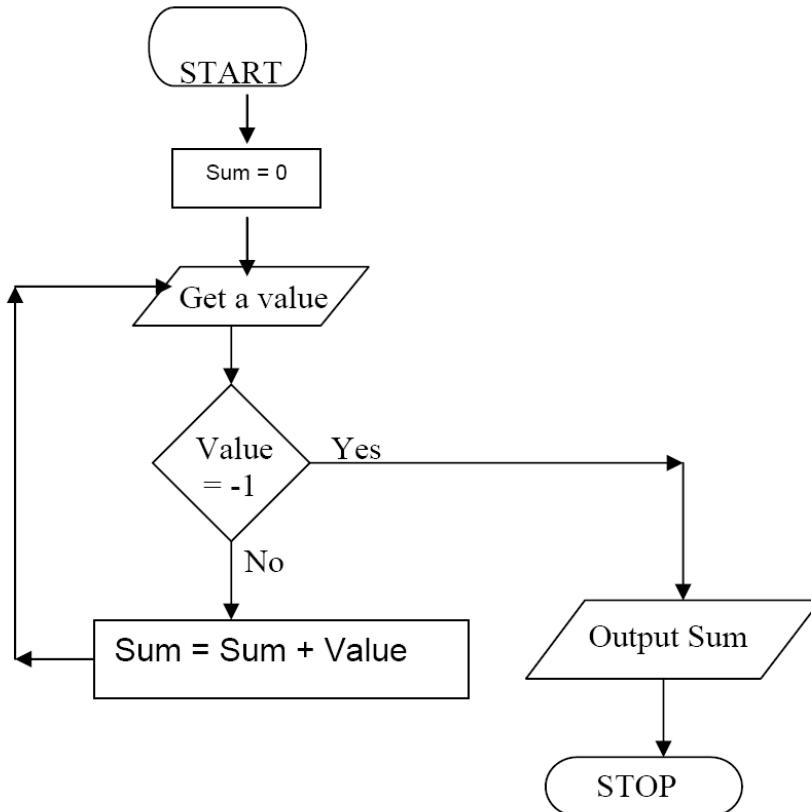
Connectors

Flowchart of finding Average of three numbers



Flowchart 2

- Given list of numbers 28, 47, 492, 9387, 48960, 2, -1. Draw a flow chart to find the sum of all the positive numbers in this list.



Pseudo code to find the average of 3 numbers



Declare Num1, Num2, and Num3 as integers

Declare Average as real

Write "Welcome to the program. Enter 3 positive numbers",

INPUT Num1, Num2, Num3

Average = (Num1 + Num2 +Num3) / 3

Write "The average of three numbers entered is",
Average

Stop

Pseudo code for finding area and circumference of circle



```
INPUT "Enter radius of circle:", r
```

```
ac = 3.14 * r * r
```

```
c = 2 * 3.14 * r
```

```
PRINT "Area of circle=", ac
```

```
PRINT "Circumference of circle=", c
```

```
END .
```

Pseudo code for finding area and circumference of circle



```
INPUT "Enter radius of circle:", r
```

```
ac = 3.14 * r * r
```

```
c = 2 * 3.14 * r
```

```
PRINT "Area of circle=", ac
```

```
PRINT "Circumference of circle=", c
```

```
END .
```

Hello World Program in C



```
#include <stdio.h>

int main() {
    /* my first program in C */
    printf("Hello, World! \n");

    return 0;
}
```

Hello World Program in C++



```
// Our first program in C++  
#include <iostream>  
  
int main()  
{  
    std::cout << "Hello World!"  
}
```



tutorialspoint.com (Online compilation)



 tutorialspoint | Hello World Example in C 

Execute | Beautify | Share | Source Code | Help

```
1 #include <stdio.h>
2
3 int main() {
4     /* my first program in C */
5     printf("Hello, World! \n");
6
7     return 0;
8 }
```

Terminal

Hello, World!

← → C

ideone.com/Ev91GK

 edit  fork  download

 copy

```
1. #include <stdio.h>
2.
3. int main(void) {
4.     printf("Hello, World! \n");
5.     return 0;
6. }
```

Success #stdin #stdout 0s 5400KB

 comments (0)

 stdin

 copy

Standard input is empty

 stdout

 copy

Hello, World!

<https://ideone.com/Ev91GK>



CODECHEF
A **Directi** Educational Initiative

Username Password Login [Forgot Password](#)

[g+](#) [Q](#) [f](#)

[PRACTICE](#) [COMPETE](#) [DISCUSS](#) | [COMMUNITY](#) [HELP](#) [ABOUT](#)


Practice
Prepare yourself for various kinds of programming challenges by solving practice problems of varying difficulty.


Compete
Compete against other techies in our monthly programming contests, gain recognition and win cool stuff. Coding contests begin on the first of every month.


Discuss
Hang out with other coders, discuss algorithms, get your questions answered and be inspired.

< Practice • July LunchTime 2016 • July Cook-Off 2016 • Performance >

New to Programming and don't know where to start? [Start Here](#)

Follow us - [f](#) [t](#) [g+](#) [yout](#) [Q](#)

Online C++ compiler



tutorialspoint | C++ Program Structure

Execute | Beautify | Share | Source Code | Help | Terminal

```
1 #include <iostream>
2 using namespace std;
3
4 // main() is where program execution begins.
5 int main() {
6     cout << "Hello World"; // prints Hello World
7     return 0;
8 }
```

Hello World

 edit  fork  download

```
1. #include <iostream>
2. using namespace std;
3.
4. int main() {
5.     cout << "Hello World";// your code goes here
6.     return 0;
7. }
```

Success #stdin #stdout 0.01s 5432KB

 stdin

Standard input is empty

<https://ideone.com/Wy7DKP>

 stdout

Hello World



Thank You!

