

Hello CPPBuddies

Day 01

Welcome

To

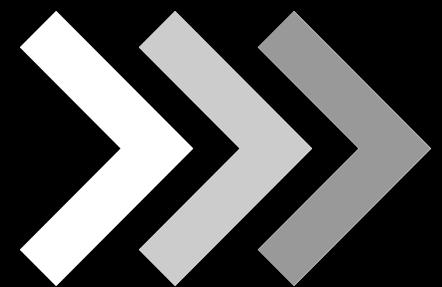
C++ COMPLETE BOOTCAMP

Your Guide To A Solid Foundation in C++

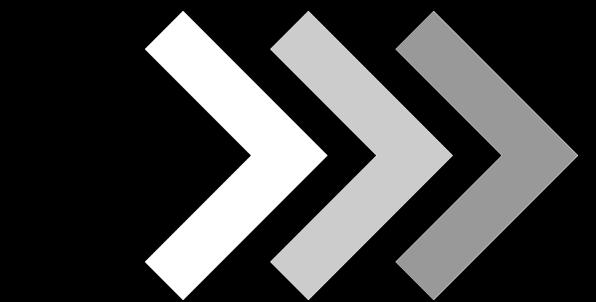
Let us begin

C++ PROGRAMMING LANGUAGE

The best language for learning programming and DSA



ACTION
TIME



C++ COMPLETE BOOTCAMP



LECTURE 01
DAY 01 WEEK 01

C++ PROGRAMMING GETTING STARTED.



```
C++ hello.cpp X
```

```
C++ hello.cpp
1 #include <iostream>
2 using namespace std;
3
4 int main(){
5     cout << "Hello World\n";
6     return 0;
7 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
AKASH MAJI@Akash320 MINGW64 ~/Desktop/CPP
$ g++ -v
Using built-in specs.
COLLECT_GCC=C:\MinGW\bin\g++.exe
COLLECT_LTO_WRAPPER=c:/mingw/bin/../libexec/gcc
Target: mingw32
Configured with: ../src/gcc-6.3.0/configure --b
mpc=/mingw --with-isl=/mingw --prefix=/mingw --
jc,obj-c++,fortran,ada --with-pkgversion='MinG
isable-sjlj-exceptions --enable-version-specific
ibstdcxx-debug --with-tune=generic --enable-lib
Thread model: win32
gcc version 6.3.0 (MinGW.org GCC-6.3.0-1)
```

```
AKASH MAJI@Akash320 MINGW64 ~/Desktop/CPP
$ git --version
git version 2.30.1.windows.1
```

```
AKASH MAJI@Akash320 MINGW64 ~/Desktop/CPP
$ []
```

Hello World Program

Basic Program
for
Beginners



<===== check g++ compiler version

How to compile
\$ g++ main.cpp -o main.exe

How to execute
\$./main.exe

<===== check git version



First Program

What You Need to Know

Understand how the first C++ program works

HELLO WORLD

The first program
of your journey.

Components

- Header files
- Main function
- Print Statement
- Return Statement

Header files **defined**

`#include <filename.h>`

or

`#include "filename.h"`

A variety of **functions** or **objects** or **constants** present in a **file** called **header file** or **library file**.

By including a **header** file, we can use its contents in our program.

Without header file, programming would be very **hard** to do.

There are of 2 types of header files:

Pre-existing header files:

Files which are already available in C/C++ compiler we just need to import them using **#include <.....>**

User-defined header files:

These files are defined by the user and can be imported using
#include ".....".

Preprocessor directives

Directives in the source file tell the preprocessor to take specific actions

#define

#include

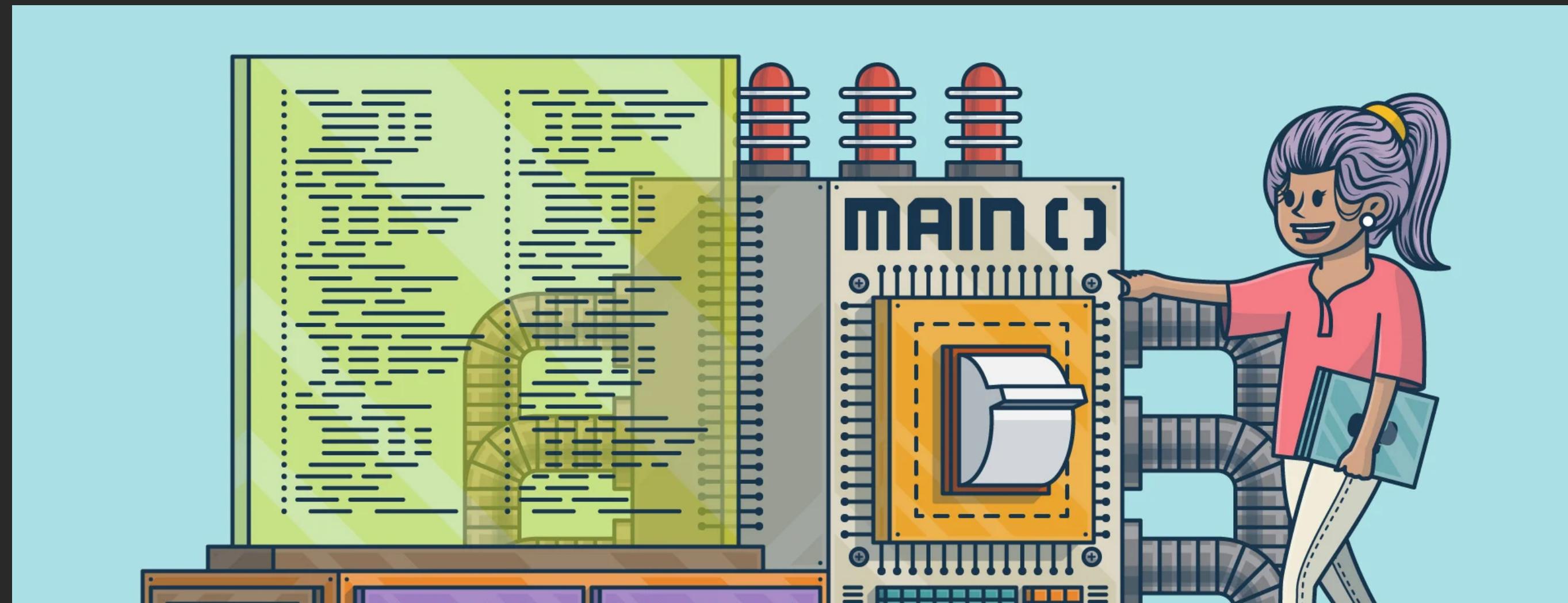
#ifndef

Why preprocessor directives?

These **preprocessor directives** are used for **instructing compiler** that these files need to be processed before compilation.



The **first function** to be **called** by the OS and **executed** by hardware



STDIN



Standard Input Device



Standard Output & Standard Error Device

STDOUT & STDERR

INPUT & OUTPUT

Interacting with the **program** is necessary to do something **useful**

standard input -> STDIN
standard output -> STDOUT
standard error -> STDERR

(Very Important)

How is I/O performed in C++ ?

In C++, **input** and **output** are performed in the form of a **sequence of bytes**

or more commonly known as **streams**

```
#include <iostream>
```

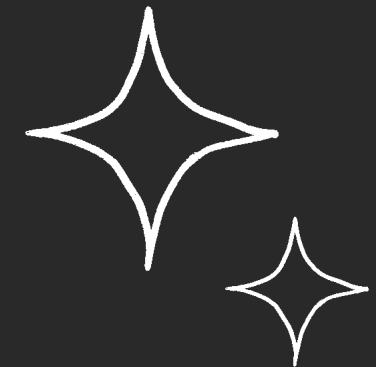


We need a library for
input & output
in C++

iostream
Input Output Stream

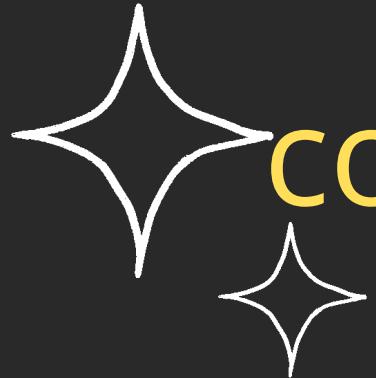
Three Objects

cin



console input => used for taking
input from STDIN

cout



console output => used for showing
output to STDOUT

cerr

console error => used for showing
error to STDERR

Older Way

(Don't Use This)



Newer Way

(Do Use This)

- #include <iostream.h>
- Used in older compilers like Turbo C/C++, Borland C/C++ Compiler
- Less Efficient

- #include <iostream>
- Used in modern compilers like DevC++, MinGW GNU Compiler
- More Efficient



Significance of return statement

return statement returns the **status code**
to the **OPERATING SYSTEM**

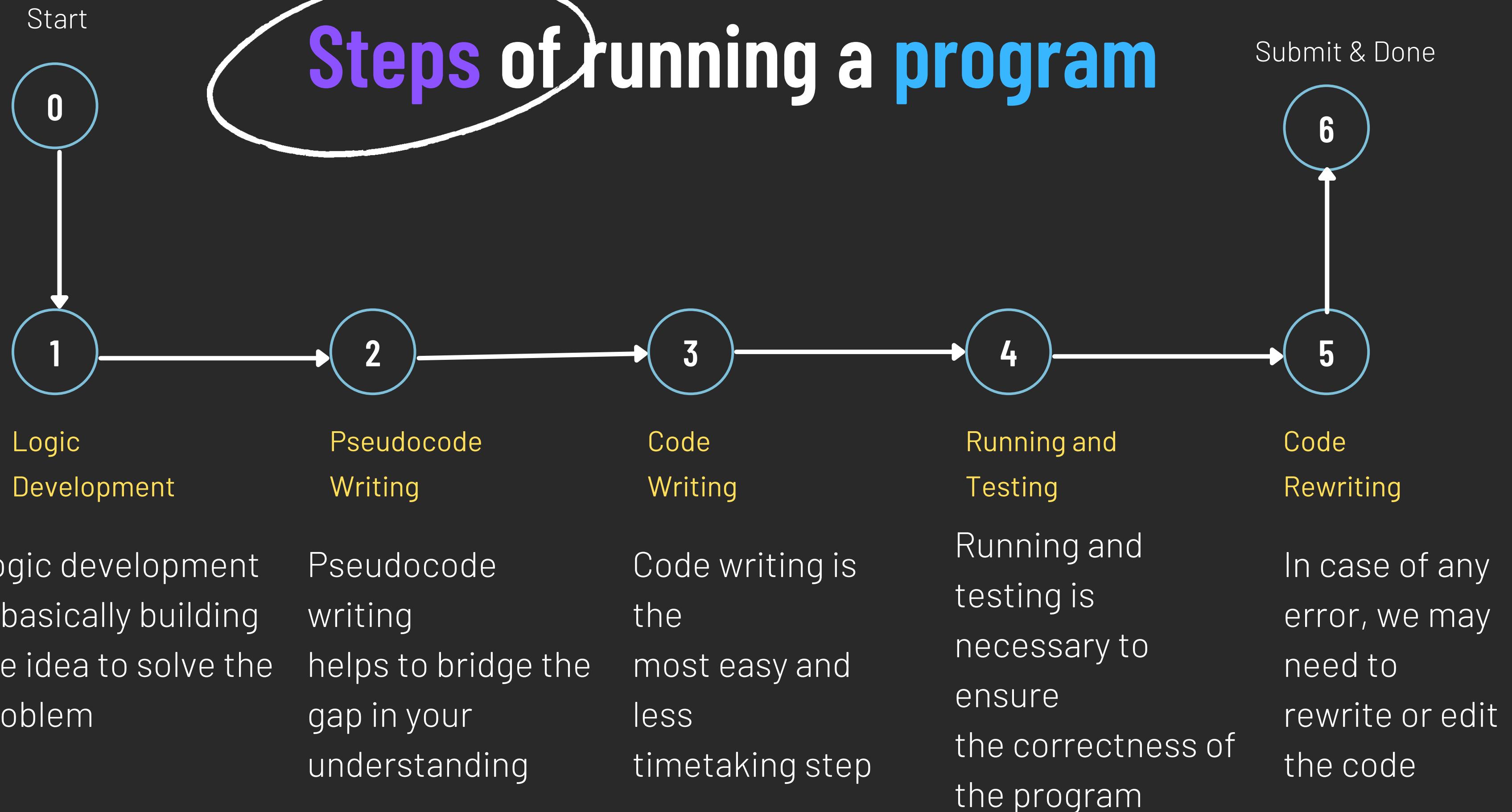
ex. return 0 return 1 return 123

0 => successful termination

1 => unsuccessful termination

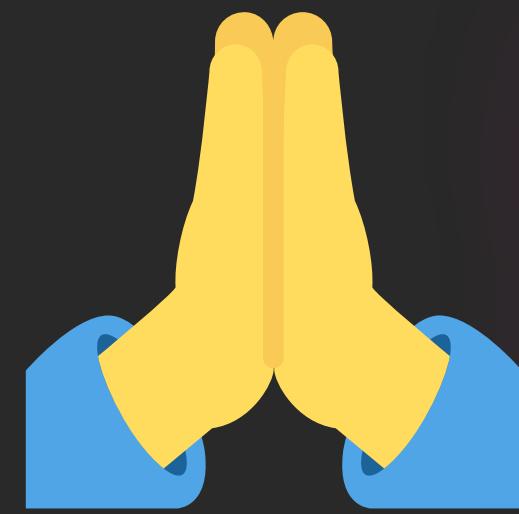


Steps of running a program

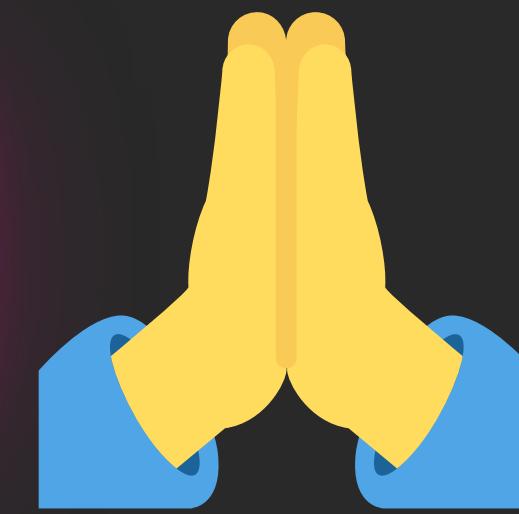




Ask Your Doubts



THANK YOU



keep calm,
wear mask,
and
study hard



whoami
AKASH MAJI
Your Mentor