Intro to Programming

C/C++

Sankalp Gupta
moklaeducation@gmail.com

Learning Goals

- Comfort with
  - Basics of C/C++
  - Basics of Computer Science
- Familiarity with IDE
  - Visual Studio Community Edition
- Thinking logically
  - i.e. One step at a time
  - And Visualizing how computer works
- Independence of technology

Why C/C++

- Foundational understanding
  - Understand Computer Science
- Speed and control
  - Fastest programming language
- Really small programming language
  - C has 32 keywords
  - C++ has 92 keywords as of 2023
- Makes you digital native

When NOT C/C++?

- Slower manual speed of writing code
- Don’t care about speed
- Don’t care about deep Computer Science
    - Although this may not be achievable

Bit : Smallest unit of memory

- Bits are used for representing everything
- Have 2 states : 0 and 1 , like a bulb
    - On : 1
    - Off : 0
- Nibble : 4 bits
- Byte : 8 bits
- int (integer) 4 bytes
- char (character) 1 byte

Hertz : Unit of time and speed in Computers

- 1 Hertz : once per second
    - 1 unit of work per clock instruction
- Modern processors
    - Measured in Giga hertz
    - High Core Counts
    - More instructions

Hello World

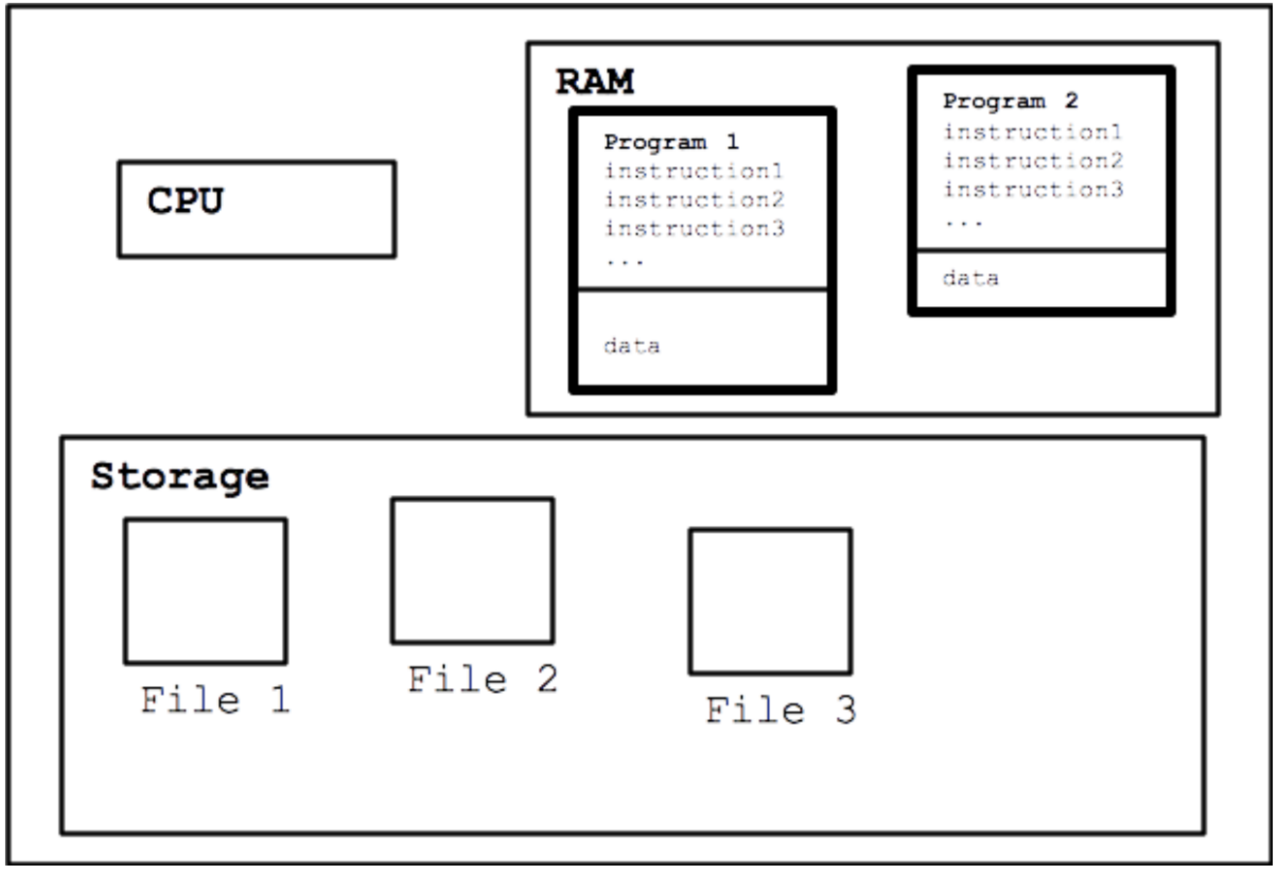
- Let’s go ,
- Program to print “Hello World”

```
#include <iostream>
using namespace std;
int main()
{
    cout << “Hello World”;
    return 0;
}
```

Print a diamond pattern

```
  *
 ***
*****
 ***
  *
```

Running the program : model



Data Types(Primitive/built in)

- bool
- int
  - short(2), long(4)
  - signed,unsigned
- float
  - float(4), double(8) , long double(8,10,16)
- char(1)
  - signed,unsigned
  - wchar_t(2)
- void

Variables

- Containers for storing data
- Value can change during execution (unless you don’t want it to)
- Declaration
    - int birthyear;
    - float weight;
    - char courseGrade;

Data Types (Derived)

- Arrays
  - char name[100];
    - Size 100
    - Index : 0 - 99
  - int age[10];
    - Size 10
    - Index : 0 - 9
  - float power[20]
    - Size 20
    - Index : 0 - 19

Integer vs floating math

- Division
  - Float :continuos, contains decimal point
  - Int : discrete , truncates everything after decimal ,
- float f = 10;
  - cout << f/3;
  - 3.33
- int i = 10;
  - cout<< i/3;
  - 3

Type Name	Bytes	Other Names	Range of Values
int	4	signed	-2,147,483,648 to 2,147,483,647
unsigned int	4	unsigned	0 to 4,294,967,295
bool	1	none	false or true
char	1	none	-128 to 127 by default
signed char	1	none	-128 to 127
unsigned char	1	none	0 to 255
short	2	short int, signed short int	-32,768 to 32,767
unsigned short	2	unsigned short int	0 to 65,535
long	4	long int, signed long int	-2,147,483,648 to 2,147,483,647
unsigned long	4	unsigned long int	0 to 4,294,967,295
long long	8	None	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807
unsigned long long	8	none	0 to 18,446,744,073,709,551,615
enum	varies	None	
float	4	None	3.4E +/- 38 (seven digits)
double	8	None	1.7E +/- 308 (fifteen digits)
long double	8	None	Same as double
wchar_t	2		0 to 65,535

User Defined Data Type

- Enum
  - Short for enumeration
  - Define a set of named, integer constants

```
enum ESpeed
{
   low,
   medium,
   high
};

ESpeed fanSpeed;
fanSpeed = low;
```

Code : Greeting with name/age

- Given user
  - Store name in char array (char name[100])
  - Store year of birth in int(int birthyear)
- Calculate age
  - age = currentYear – birthyear;
- Print Name, age.
  - cout <<“Hello “

    ```
    << name
    <<“ I know your age is :“
    <<age;
    ```

Advanced : Count digits in a number

- Define a number
- Return the number of digits, (assume positive)
  - 129 → 3
  - 34 → 2
  - 0 → 1
- Hints
  - You need to know loops
    - while
  - You need to know integer maths
    - division

Input , Output and Processing for Humans

- Speech
  - Process →Speak →Listen
- Book
  - Read →Process → Memorize
- Conversation
  - Listen → Speak →Listen →Speak ….

I/O and processing for Computers

- Video games
  - Input (controllers) →Process →Output (Screen)
- Movies
  - Input (network) →Process → Output (visuals, audio)
- Console
  - Input(char, int, float) → Process →Output (char(s), int, float)

Standard Input

- cin
  - Read data from keyboard
  - Store it in variables
- Extraction operator
  - >>
- Can use multiple data types (char, int, float , ...)
- Example
  - int age;
  - cin >> age;

Standard Output

- cout
  - Write data to console/Screen
  - Reads from memory
- Insertion operator
  - <<
- Can use multiple data types(variables, literals, constants)
- Example
  - int age = 172;
  - cout << age;

Operators >>,<<

- Extraction and Insertion operators
- Can be cascaded
  - cin >>age >> name;
  - cout<<age << name;
- << works with stream modifiers
  - “\n” : newline
    - cout <<“\n”; //moves the cursor to new line
  - Or
    cout <<endl; //Same visual effect as “\n” but is different
  - (there are other stream modifiers too)

Lab

- Using Cin (Hint: define a variable first)
  - Input a character (char)
  - Input an integer(int)
  - Input a decimal(float)
- Using cout
  - Output a character
  - Output an integer
  - Output a decimal.
- Use endl and “\n”
- Cascade the operators

string data type

- We know
  - Int , float , char
- string
  - Derived data type,
  - Useful for names, description etc.
- Header
  - #include<string>
- Usage
  - string name;
  - cin >> name;
  - cout <<name;

Assignment : Mad libs story

```
 îŋçľuđê  îộṣʧsêắṇ      Rêṛuîsêđ ĝộs îŋřuʧ ộuʧřuʧ ộřêṣắʧîộŋṣ  çộuʧ  çîŋ
 îŋçľụđê  ṣʧsîŋĝ       Ṛêṛụîsêđ ĝộs ụṣîŋĝ ṣʧsîŋĝ đắʧắ ʧỳřê
îŋʧ ṇắîŋ
        ṣʧđ  ṣʧsîŋĝ ắđkêçʧîŵê,  ŋộụŋ,  ŵêsč,  ắđkêçʧîŵê,  ŋộụŋ,
           Rsộŋřʧ ʧḥê uṣêṣ ĝộs îŋřuʧ ắŋđ ṣʧộsê îʧ îŋ ʧḥê ŵắsîắčľêṣ
        ṣʧđ  çộuʧ     Éŋʧêṣ ắŋ ắđkêçʧîŵê
        ṣʧđ  çîŋ    ắđkêçʧîŵê,

        ṣʧđ  çộuʧ     Éŋʧêṣ ắ ŋộụŋ
        ṣʧđ  çîŋ    ŋộụŋ,

        ṣʧđ  çộuʧ     Éŋʧêṣ ắ ŵêsč
        ṣʧđ  çîŋ    ŵêsč,

        ṣʧđ  çộuʧ     Éŋʧêṣ ắŋộʧḥêṣ ắđkêçʧîŵê
        ṣʧđ  çîŋ    ắđkêçʧîŵê,

        ṣʧđ  çộuʧ     Éŋʧêṣ ắŋộʧḥêṣ ŋộụŋ
        ṣʧđ  çîŋ    ŋộụŋ,

           Cộŋṣʧsụçʧ ắŋđ řsîŋʧ ʧḥê Ňắđ Ľîčṣ sʧộsỳ
        ṣʧđ  çộuʧ     ŋ    Ÿộus Ňắđ Ľîčṣ Şʧộsỳ,  ŋ
        ṣʧđ  çộụʧ     Ộŋçê uřộŋ ắ ʧîṇê  ʧḥêṣê xắṣ ắ     ắđkêçʧîŵê,         ŋộụŋ,
                      Iʧ ľộŵêđ ʧộ     ŵêṣč,      ắľľ đắỳ ľộŋĝ  ŋ
                    Ộŋê đắỳ  îʧ ŋêʧ ắ     ắđkêçʧîŵê,         ŋộụŋ,
                      ắŋđ ʧḥêỳ ľîŵêđ ḥắřřîľỳ êŵêṣ ắĝʧêṣ  ŋ

        sêʧụsŋ ˳     Íŋđîçắʧê ṣụççêṣṣĝụľ řsộĝsắṇ êỵêçuʧîộŋ
```

Boss Assignment

- Input student details
    - Student Name
    - Subject name
    - Marks (out of 100)
- Process
    - Find grade using this table
    - 90 <marks →A
    - 75 <= marks < 90 →B
    - 60 <= marks <=74 →C
    - marks <60 →D
- Output
    - Grade for the student
- Challenge
    - Enter multiple students, print how many students had A, B , C and D grades each.
- Hint
    - Need to know conditional (if-else)
    - May need to know loop (while)

Conditionals

- Are you going to store?
  - Yes
- If they have chocolate, get two , if not then get icecream
  - Ok.
- Code

  -
    ```
    if( haveChocolate)
    {
        //Buy 2 chocolate
    } else
    {
       //Buy icecream
    }
    ```