C++ Notes

What is Function?

Defination , Declaration , and call

Declaration

Ways to define arguments

Types of formal arguments

Call by value ,call by address and call by reference

What is function?

Function is block of code performing a unit task.

Function has a name , return type and arguments.

Function is way to achieve modularization.

Function are predefine and user defined.

Pre-defined are declared in header files and defined in library files

Defination Declaration and call

#include<iostream.h> //declaration of cout and cin

Void main()

{

Void fun() //function declaration

Cout<<”you are in main”;

Fun() //function call local declare

}

Void fun() //function definition

{

Cout<<”you are in fun”;

}

Function declaration is also known as function prototype

Function needs to be declared before use ( just like variables)

Function can be declared locally or globally

Return type function name ( argument list);

Function defination is a block of code.

Ways to define a function

Takes nothing returns nothing

Takes something return nothing

Takes nothing return something

Takes something return something

Formal and actual arguments

#include<iostream.h>

Int sum(int , int)

Void main()

{

Int a=5, b=6;

Int s= sum(a,b); a and b are actual argument

Cout<<”sum is “<<s;

}

Int sum(int x ,int y) x and y are formal argument

{

Return(x+y);

}

Types of formal arguments

Ordinary variable of any type

Pointer variables

Reference variables

Call by value when formal argument are ordinary variable it is function call by value

#include<iostream.h>

Int sum(int ,int);

Void main()

{

Int a=5 , b=6;

Int s=sum(a,b);

Cout <<”sum”<<s;

}

Int sum (int x,int y)

{

Return(x+y);

}

Call by address

#include<iostream.h> // when formal arguments are pointer variables it is function call by address

Int sum(int x ,int y);

Void main()

{

Int a=5,b=6;

Int s=sum(&a,&b);

Cout <<”sum is “<<s;

}

Int sum (int\*p,int\*q)

{

Return(\*p+\*q);

}