***AssignmentNo:-1***

1

Name :- Mujtaba Ahmed

Roll:CSU-F17-128

Subject:- OOP

**C++ Syntax and one short example of the following**

**Declaration of variables**

C++ is a strongly-typed language, and requires every variable to be declared with its type before its first use. This informs the compiler the size to reserve in memory for the variable and how to interpret its value. The syntax to declare a new variable in C++ is straightforward: we simply write the type followed by the variable name (i.e., its identifier). For example.

|  |  |
| --- | --- |
| 1 2 | int a;  float mynumber; |

* **1- Data types :-**

There are three basic data types associated with variables:

1. **int** - integer: a whole number.
2. **float** - floating point value: ie a number with a fractional part.
3. **char** - a single character.

**Int:-**

**Syntax:-**

Declartion data type and variable: (int a;)

**Example :**

#include <iostream>

using namespace std;

int main ()

{

// declaring variables:

int a, b;

int result;

// process:

a = 5;

b = 2;

a = a + 1;

result = a - b;

// print out the result:

cout << result;

}

**Float:-**

**Syntax:-**

Declartion data type and variable: (float a;)

**Example :**

#include <iostream>

using namespace std;

int main ()

{

// declaring variables:

float a, b;

float result;

// process:

a = 5.3;

b = 2.5;

a = a + 1;

result = a - b;

//print out the result:

cout << result;

}

**char:-**

**Syntax:-**

Declartion data type and variable: (char a;)

**Example :**

#include <iostream>

using namespace std;

int main ()

{

char a,flag;

flag=a;

cout<<flag;

}

* **2- Array :-**

**Array can be divided into following types:**

1. [**One Dimensional Array**](https://www.programtopia.net/cplusplus/docs/arrays#one-dimension)
2. [**Multi Dimensional Array**](https://www.programtopia.net/cplusplus/docs/arrays#multi-dimension)

## One Dimensional Array

An array in which data are arranged linearly in only one dimension is called one dimensional array. It is commonly known as **1-D array**

### Syntax and Declaration of One Dimensional Array

datatype array\_name[size];

**Example :**

#include <iostream>

#include <conio.h>

using namespace std;

int main()

{

int arr[10],sum=0,i;

cout<<"Enter 10 numbers"<<endl;

for(i=0;i<10;i++)

{

cin>>arr[i];

sum = sum+arr[i];

}

cout<<"Sum = "<<sum;

getch();

return 0;

}

### Two Dimensional Array:

Two dimensional array is where the data is stored in a list containing 2-D array.

**Syntax and Declaration of Two Dimensional Array**

datatype array\_name[d1][d2];

**Example :**

#include <iostream>

#include <conio.h>

using namespace std;

int main()

{

int arr[10][10],row,col,i,j;

cout<<"Enter size of row and column: ";

cin>>row>>col;

cout<<"Enter elements of matrices(row wise)"<<endl;

for(i=0;i<row;i++)

for(j=0;j<col;j++)

cin>>arr[i][j];

cout<<"Displaying matrix"<<endl;

for(i=0;i<row;i++)

{

for(j=0;j<col;j++)

cout<<arr[i][j]<<" ";

cout<<endl;

}

getch();

return 0;

}

* **Loop Operations (for, while, Do-While and switches)**

**There are 3 type of loops in C++ Programming**:

* [**for Loop**](https://www.programiz.com/cpp-programming/for-loop)
* **while Loop**
* **do...while Loop**
* [**for Loop**](https://www.programiz.com/cpp-programming/for-loop)**:-**

## C++ for Loop Syntax

for(initializationStatement; testExpression; updateStatement) {

// codes

}

**Example :**

#include <iostream>

using namespace std;

int main()

{

int i, n, factorial = 1;

cout << "Enter a positive integer: ";

cin >> n;

for (i = 1; i <= n; ++i) {

factorial \*= i; // factorial = factorial \* i;

}

cout<< "Factorial of "<<n<<" = "<<factorial;

return 0;

}

## C++ while Loop:-

The syntax of a while loop is:

while (testExpression)

{

// codes

}

**Example :**

#include <iostream>

using namespace std;

int main()

{

int number, i = 1, factorial = 1;

cout << "Enter a positive integer: ";

cin >> number;

while ( i <= number) {

factorial \*= i; //factorial = factorial \* i;

++i;

}

cout<<"Factorial of "<< number <<" = "<< factorial;

return 0;

}

## C++ do...while Loop

The do...while loop is a variant of the while loop with one important difference. The body of do...while loop is executed once before the test expression is checked.

The syntax of do..while loop is:

do {

// codes;

}

while (testExpression);

**Example :**

#include <iostream>

using namespace std;

int main()

{

float number, sum = 0.0;

do {

cout<<"Enter a number: ";

cin>>number;

sum += number;

}

while(number != 0.0);

cout<<"Total sum = "<<sum;

return 0;

}

## C++ switch...case syntax

switch (n)

​{

case constant1:

// code to be executed if n is equal to constant1;

break;

case constant2:

// code to be executed if n is equal to constant2;

break;

.

.

.

default:

// code to be executed if n doesn't match any constant

}

**Example :**

#include <iostream>

using namespace std;

int main()

{

char o;

float num1, num2;

cout << "Enter an operator (+, -, \*, /): ";

cin >> o;

cout << "Enter two operands: ";

cin >> num1 >> num2;

switch (o)

{

case '+':

cout << num1 << " + " << num2 << " = " << num1+num2;

break;

case '-':

cout << num1 << " - " << num2 << " = " << num1-num2;

break;

case '\*':

cout << num1 << " \* " << num2 << " = " << num1\*num2;

break;

case '/':

cout << num1 << " / " << num2 << " = " << num1/num2;

break;

default:

cout << "Error! operator is not correct";

break;

}

return 0;

}

**Syntax of Function**

return\_type function\_name (parameter\_list)

{

//C++ Statements

}

## A simple function example

#include <iostream>

using namespace std;

/\* This function adds two integer values

\* and returns the result

\*/int

sum(int num1, int num2){

int num3 = num1+num2; return num3;

}

int main(){

//Calling the function

cout<<sum(1,99);

return 0;

}

**By using Function Calling:**

**Example:-**

#include <iostream>

using namespace std;

//Function declaration

int sum(int,int);

//Main function

int main(){

//Calling the function

cout<<sum(1,99);

return 0;

}

/\* Function is defined after the main method

\*/

int sum(int num1, int num2){

int num3 = num1+num2;

return num3;

}

**Declaring pointers**

int \* number;

char \* character;

double \* decimals;

**Example:-**

#include <iostream>

using namespace std;

int main ()

{

int firstvalue, secondvalue;

int \* mypointer;

mypointer = &firstvalue;

\*mypointer = 10;

mypointer = &secondvalue;

\*mypointer = 20;

cout << "firstvalue is " << firstvalue << '\n';

cout << "secondvalue is " << secondvalue << '\n';

return 0;

}