C++ Assingment

1. Write a class Student with a default constructor that initializes the student's name to "Unknown" and age to 0. Add a method display to print the student's details.

Understanding of code

1. First we have to create a class name student.
2. Then we have to initialize two variable first is name with string type and another one is age.
3. Now we create a default constructor. And in this we have to put the value for name and age.
4. In this class we have to create an function called display to display the name and age.
5. In the main() function we have to display and call that function.

Code :

#include <iostream>

#include<string>

using namespace std;

class Student{ //create a class name Student

private:

string name;

int age ;

public :

Student(){ //default constructor

name = "Unknown";

age = 0;

}

void display(){ // method to display the details

std::cout << "Name : " <<name<<" " << "Age : "<<age<< std::endl;

}

};

int main()

{

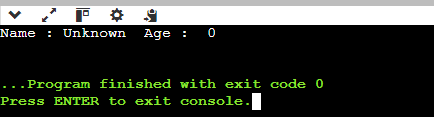
Student s;

s.display(); // calling the function in class to display the details

return 0;

}

Output :



1. Write a class Rectangle with a parameterized constructor that initializes the length and width. Add a method area that returns the area of the rectangle.

Understanding of code :

1. First we have to create a class name rectangle.
2. Then we have to take two variable which type is double and name is length and width.
3. Then we create a parameterized constructor .
4. And now we have to create a one function name area to get the area of rectangle.
5. In the main() function we create an instance of class name rec .
6. With this instance we have to call that function inside in class to print area.
7. In main() function we create one more variable for display the area.

Code :

#include <iostream>

class Reactangle{ // create class

private:

double len;

double wid;

public:

Reactangle(double l, double w){ // parameterized constructor

len = l;

wid = w;

}

double area(){ // function for getting area

return len \* wid;

}

};

int main()

{

Reactangle rec(4.0,5.0); // give arugment

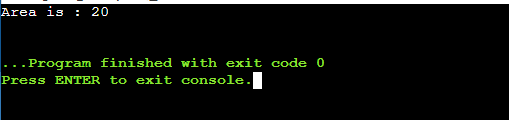
double result = rec.area(); // call the function and get the value in result

std::cout <<"Area is : "<< result << std::endl;

return 0;

}

Output :



1. Write a class Book that has both a default constructor and a parameterized constructor. The default constructor should set the title to "Unknown" and the number of pages to 0. The parameterized constructor should initialize the title and pages with given value.

Understanding the code :

1. In this we have to create two constructor.
2. Default and Parametrized both the constructor we have to create .
3. In main() we have to create two instance of the class name Book (b1 and b2).
4. B1 is call function that we have to create in class for display.
5. B2 is also call function but with some arguments.

Code :

#include <iostream>

#include <string>

using namespace std;

class Book{ // create class

private:

string title ;

int pages;

public :

Book(){ // default constructor

title = "Unknown";

pages = 0;

}

Book(string n, int p){ // parameterized constructor

title = n;

pages = p;

}

void display(){ // display the title and pages

std::cout <<"Title : "<< title <<" Pages : "<<pages<< std::endl;

}

};

int main()

{

Book b1; // create instance

b1.display(); // call function

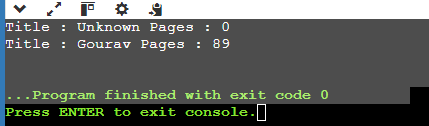
Book b2("Gourav", 89); //create another instance with argument

b2.display(); // call function

return 0;

}

Output :



1. Write a class Complex that represents complex numbers. Implement a default constructor that sets both real and imaginary parts to 0, and a parameterized constructor that takes two arguments to initialize the real and imaginary parts

Understanding the code :

1. In this we have to create two constructor.
2. Default and Parametrized both the constructor we have to create.
3. In main() we have to create two instance of the class name Complex(c1 and c2).
4. C1 is call function that we have to create in class for display.
5. C2 is also call function but with some arguments.

Code :

#include <iostream>

class Complex{ // create class

private:

double real ;

double imaginary;

public :

Complex(){ // default constructor

real = 0.0;

imaginary = 0.0;

}

Complex(double r, double i){ // parameterized constructor

real = r;

imaginary = i;

}

void display(){ // display the real and imaginary

std::cout <<"real : "<< real <<" imaginary : "<<imaginary<< std::endl;

}

};

int main()

{

Complex c1; // create instance

c1.display(); // call function

Complex c2(2.5, -1.5); //create another instance with argument

c2.display(); // call function

return 0;

}

Output :



1. Write a function increment that takes a pointer to an integer and increments its value by 1. Demonstrate the function in the main program.

Understanding the code

1. First we take an variable name num and give the value in it ;
2. Then we call the function name increment give argument is address of num.
3. Function increment the value and then print.

Code :

#include <iostream>

int Increment(int \*ptr){ // function for increment

return (\*ptr)++;

}

int main()

{

int num = 10;

std::cout << "Before Increment : "<< num;

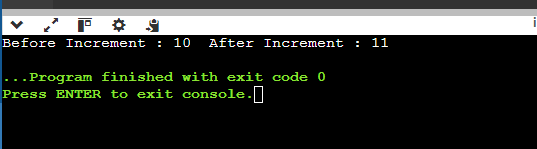
Increment(&num); // call the function

std::cout << " After Increment : "<< num;

return 0;

}

Output :



1. Write a class Circle with a method area. Create a pointer to an object of this class and call the area method using the pointer.

Understanding the code :

1. In this we have to create two constructor.
2. Default and Parametrized both the constructor we have to create.
3. In main() we have to create two pointer instance of the class name Circle.
4. We have to create one function for calculating the area of circle .

Code :

#include <iostream>

class Circle{ // create class

private:

double radius ;

public :

Circle(double r) { // parameterized constructor

radius = r;

}

double area() { // function for getting area of circle

return 3.14 \* radius \* radius; // 3.14 is pi value

}

};

int main() {

Circle \*circlePointer; // create a pointer instance of class

circlePointer = new Circle(5.0); // memory allocation

double circleArea = circlePointer->area(); // call function and take value in circleArea variable std::cout << "The area of the circle with radius " << circlePointer->area() << " is " << circleArea;

return 0;

}

Output :



1. Write a program that creates an array of pointers to integers. Initialize the array with values and print them using the pointers.

Understanding the code :

1. First we take a constant SIZE variable value outside the main() function.
2. Now we have to create an pointer array with sizeof (SIZE) ;
3. Then we just create 5 variable but diff vale and give all the variable address to pointer array.
4. Then now for loop to print value

Code :

#include<iostream>

using namespace std;

const int SIZE = 5; // constant value of SIZE variable

int main(){

int \*ptrArr[SIZE]; // crate a pointer array

int v1 = 5, v2=10,v3=15,v4=20,v5=25;

ptrArr[0] = &v1; // give pointer array adress of v1 to v5 for 0 to 4 index

ptrArr[1] = &v2;

ptrArr[2] = &v3;

ptrArr[3] = &v4;

ptrArr[4] = &v5;

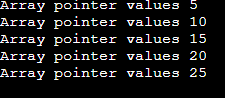
for(int i=0;i<5;i++){ // for loop for printing value

std::cout <<"Array pointer values " <<\*ptrArr[i] << std::endl;

}

}

Output :



1. Write a function that takes a pointer to an array of integers and the size of the array. The function should print all elements of the array

Understanding the code :

1. First we have to create an array with values.
2. Than we find size of array with sizeof .
3. Then call the function name printArray .
4. This function takes array as an pointer argument .

Code :

#include<iostream>

using namespace std;

void printArray(int \*arr, int s){ // function to print array takes arrgument pointer

for(int i=0;i<s;i++){ // for loop for print

std::cout << arr[i] << std::endl;

}

}

int main(){

int array[] = {10,20,30,40,50};

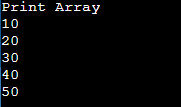
int size = sizeof(array)/sizeof(array[0]); // this is to find the size of array (sizeof-> it gives us bytes)

std::cout << "Print Array" << std::endl;

printArray(array, size); // call the function

}

Output :



1. Write a function swap that takes two integer references and swaps their values. Demonstrate the function in the main program

Understanding the code :

1. First we have to get two variable .
2. Then call the swap function .
3. This function takes refrence argument .
4. And swap the value .

Code:

#include <stdio.h>

void swap(int &a, int &b) { // function takes refrence arguments

int temp = a;

a = b;

b = temp;

}

int main() {

int a = 10, b = 20;

printf("a = %d, b = %d\n", a, b);

swap(a, b); // calling function

printf(" a = %d, b = %d\n", a, b);

return 0;

}

Output :

