Easy programs

A.Ganesh Reddy

192110023

1

#include<iostream>

using namespace std;

int main()

{

cout <<"Hello world";

}

A screenshot of a computer

Description automatically generated

2.

#include<iostream>

int add(int x,int y)

{

return x+y;

}

using namespace std;

int main()

{

int x,y,z;

cout <<"enter the value of x=";

cin >>x;

cout <<"enter the value of y=";

cin >>y;

z=add(x,y);

cout <<z;

}

A screenshot of a computer

Description automatically generated

3.

#include<iostream>

using namespace std;

void swap(int \*p,int \*q)

{

int temp;

temp=\*p;

\*p=\*q;

\*q=temp;

cout <<"after swapping \n";

cout <<\*p<<"\n"<<\*q;

}

int main()

{

int x=3,y=4;

swap(&x,&y);

}

A screenshot of a computer

Description automatically generated

4.

#include <iostream>

using namespace std;

int modifyInteger(int& value) {

value += 10;

return value;

}

int main()

{

int number = 5;

cout << "Initial value of number: " << number << endl;

modifyInteger(number);

cout << "Modified value of number: " << number << endl;

return 0;

}

A screenshot of a computer

Description automatically generated

5.

#include <iostream>

inline int square(int x) {

return x \* x;

}

int main() {

int number;

std::cout << "Enter an integer: ";

std::cin >> number;

int squaredNumber = square(number);

std::cout << "The square of " << number << " is: " << squaredNumber << std::endl;

return 0;

}

A screenshot of a computer

Description automatically generated

6.

#include <iostream>

#include <string>

void printMessage(std::string message = "Hello") {

std::cout << message << std::endl;

}

int main() {

printMessage();

return 0;

}

A screenshot of a computer

Description automatically generated

7.

#include<iostream>

using namespace std;

void area(float l,float b)

{

cout <<"area of rectangle="<<l\*b<<endl;

}

void area(float r)

{

double pi=3.14;

cout <<"area of circle="<<pi\*r\*r<<endl;

}

int main()

{

int l,b,r;

cout <<"enter the length and breadth\n";

cin >>l>>b;

cout <<"enter the radius=";

cin >>r;

area(l,b);

area(r);

}

A screenshot of a computer program

Description automatically generated

8.

#include <iostream>

class Rectangle {

private:

double length;

double width;

public:

Rectangle(double len = 0.0, double wid = 0.0) : length(len), width(wid) {}

void setLength(double len) {

length = len;

}

void setWidth(double wid) {

width = wid;

}

double getLength() const {

return length;

}

double getWidth() const {

return width;

}

};

int main() {

Rectangle rect;

rect.setLength(5.0);

rect.setWidth(3.0);

std::cout << "Length of the rectangle: " << rect.getLength() << std::endl;

std::cout << "Width of the rectangle: " << rect.getWidth() << std::endl;

return 0;

}

A screenshot of a computer

Description automatically generated

9.

#include <iostream>

#include <array>

class Rectangle {

private:

double length;

double width;

std::array<int, 2> coordinates;

public:

Rectangle(double len = 0.0, double wid = 0.0, int x = 0, int y = 0) : length(len), width(wid) {

coordinates[0] = x;

coordinates[1] = y;

}

void setLength(double len) {

length = len;

}

void setWidth(double wid) {

width = wid;

}

void setCoordinates(int x, int y) {

coordinates[0] = x;

coordinates[1] = y;

}

double getLength() const {

return length;

}

double getWidth() const {

return width;

}

std::array<int, 2> getCoordinates() const {

return coordinates;

}

};

int main()

{

Rectangle rect(5.0, 3.0, 2, 3);

std::cout << "Length of the rectangle: " << rect.getLength() << std::endl;

std::cout << "Width of the rectangle: " << rect.getWidth() << std::endl;

std::array<int, 2> coords = rect.getCoordinates();

std::cout << "Coordinates of the rectangle: (" << coords[0] << ", " << coords[1] << ")" << std::endl;

return 0;

}

10

#include <iostream>

#include <array>

class Rectangle {

private:

double length;

double width;

std::array<int, 2> coordinates;

static int totalRectangles;

public:

Rectangle(double len = 0.0, double wid = 0.0, int x = 0, int y = 0) : length(len), width(wid) {

coordinates[0] = x;

coordinates[1] = y;

totalRectangles++;

}

~Rectangle() {

totalRectangles--;

}

void setLength(double len) {

length = len;

}

void setWidth(double wid) {

width = wid;

}

void setCoordinates(int x, int y) {

coordinates[0] = x;

coordinates[1] = y;

}

double getLength() const {

return length;

}

double getWidth() const {

return width;

}

std::array<int, 2> getCoordinates() const {

return coordinates;

}

static int countRectangles() {

return totalRectangles;

}

};

int Rectangle::totalRectangles = 0;

int main() {

Rectangle rect1(5.0, 3.0, 2, 3);

Rectangle rect2(4.0, 2.0, 1, 1);

std::cout << "Total number of rectangles: " << Rectangle::countRectangles() << std::endl;

return 0;

}