

Домашнее задание к модулю «Объектно-ориентированное программирование»

Цель домашнего задания: научиться создавать классы, наследовать классы, перегружать операторы, создавать шаблонные классы, работать с контейнерами стандартной библиотеки

Задание №1: Создайте класс Rectangle с двумя закрытыми полями width и height, а также методами для вычисления площади и периметра прямоугольника.

1. Определите класс Rectangle с закрытыми полями width и height.

```
ConsoleApplication1.cpp  [X]
+ ConsoleApplication1  Rectangle
1  #include <iostream>
2  using namespace std;
3
4  // Определение класса Rectangle
5  class Rectangle {
6  private:
7      // Закрытые поля width и height
8      double width;
9      double height;
```

2. Создайте конструктор с двумя параметрами для инициализации полей width и height.

```
11 public:
12     // Конструктор с двумя параметрами для инициализации полей width и height
13     Rectangle(double w, double h) {
14         width = w;
15         height = h;
16     }
```

3. Определите метод area() для вычисления площади прямоугольника.

```
18     // Метод для вычисления площади прямоугольника
19     double area() {
20         return width * height;
21     }
```

4. Определите метод perimeter() для вычисления периметра прямоугольника.

```

23 // Метод для вычисления периметра прямоугольника
24 double perimeter() {
25     return 2 * (width + height);
26 }
27 };

```

- В функции main() запросите у пользователя длину и ширину прямоугольника, а затем создайте объект класса Rectangle с введенными значениями ширины и высоты.

```

29 int main() {
30     // Переменные для хранения значений ширины и высоты прямоугольника
31     double w, h;
32
33     // Запрос у пользователя значений ширины и высоты прямоугольника
34     cout << "Enter the width of the rectangle: ";
35     cin >> w;
36     cout << "Enter the height of the rectangle: ";
37     cin >> h;
38
39     // Создание объекта класса Rectangle с введенными пользователем значениями ширины и высоты
40     Rectangle rect(w, h);

```

- Вычислите площадь и периметр прямоугольника, используя методы area() и perimeter(), и выведите результаты вычислений на экран

```

42 // Вычисление и вывод площади и периметра прямоугольника
43 cout << "Area: " << rect.area() << endl;
44 cout << "Perimeter: " << rect.perimeter() << endl;

```

- В итоге у вас должна получиться следующая программа:

```

ConsoleApplication1.cpp
1 #include <iostream>
2 using namespace std;
3
4 // Определение класса Rectangle
5 class Rectangle {
6 private:
7     // Закрытые поля width и height
8     double width;
9     double height;
10
11 public:
12     // Конструктор с двумя параметрами для инициализации полей width и height
13     Rectangle(double w, double h) {
14         width = w;
15         height = h;
16     }
17
18     // Метод для вычисления площади прямоугольника
19     double area() {
20         return width * height;
21     }
22
23     // Метод для вычисления периметра прямоугольника
24     double perimeter() {
25         return 2 * (width + height);
26     }
27 };
28

```

```
29 int main() {
30     // Переменные для хранения значений ширины и высоты прямоугольника
31     double w, h;
32
33     // Запрос у пользователя значений ширины и высоты прямоугольника
34     cout << "Enter the width of the rectangle: ";
35     cin >> w;
36     cout << "Enter the height of the rectangle: ";
37     cin >> h;
38
39     // Создание объекта класса Rectangle с введенными пользователем значениями ширины и высоты
40     Rectangle rect(w, h);
41
42     // Вычисление и вывод площади и периметра прямоугольника
43     cout << "Area: " << rect.area() << endl;
44     cout << "Perimeter: " << rect.perimeter() << endl;
45
46     return 0;
47 }
```

Запустите и протестируйте программу. Отправьте преподавателю скриншот консоли с результатом выполнения программы.