

**Московский авиационный институт  
(Национальный исследовательский университет)**

Факультет: «Информационные технологии и прикладная математика»

Кафедра: 806 «Вычислительная математика и программирование»

Дисциплина: «Объектно-ориентированное программирование»

**Лабораторная работа № 1**

Тема: Программирование классов на языке C++

Студент: Бронников М. А.

Группа: 80-204

Преподаватель: Чернышов Л.Н.

Дата: 02.10.2018

Оценка:

Москва, 2018

## 1. Постановка задачи

Вариант 1 (Треугольник, Квадрат, Прямоугольник)

**Целью лабораторной работы является:**

- Программирование классов на языке C++
- Управление памятью в языке C++
- Изучение базовых понятий ООП.
- Знакомство с классами в C++.
- Знакомство с перегрузкой операторов.
- Знакомство с дружественными функциями.
- Знакомство с операциями ввода-вывода из стандартных библиотек.

### **Задание**

Необходимо спроектировать и запрограммировать на языке C++ классы фигур, согласно вариантов задания.

Классы должны удовлетворять следующим правилам:

- Должны иметь общий родительский класс Figure.
- Должны иметь общий виртуальный метод Print, печатающий параметры фигуры и ее тип в стандартный поток вывода cout.
- Должны иметь общий виртуальный метод расчета площади фигуры – Square.
- Должны иметь конструктор, считывающий значения основных параметров фигуры из стандартного потока cin.
- Должны быть расположены в отдельных файлах: отдельно заголовки (.h), отдельно описание методов (.cpp).

Программа должна позволять вводить фигуру каждого типа с клавиатуры, выводить параметры фигур на экран и их площадь.

## 2. Решения задачи

Программа позволяет вводить фигуру каждого типа с клавиатуры, выводит параметры фигур на экран и их площадь ( в моём варианте фигурами являются трапеция, ромб и пятиугольник).

Этапы создания программы:

- 1) создание библиотеки Figure.h с родительским классом Figure. В этом классе есть две виртуальных метода: Print (печатающий параметры фигуры и ее тип в стандартный поток вывода cout) и Square (рассчитывает площади фигур), а также конструктор.

- 2) Создание библиотек с описанием классов фигур ( в моём случае, восьмиугольника, ромба и пятиугольника); файлов .cpp (в которых описаны расчёты площади и печати параметров фигуры, три конструктора деструкторы)
- 3) Написание программы main.cpp, в которой реализовано диалоговое меню с возможностью создания (динамического) любой фигуры по варианту, вычисление её площади и вывода её на экран с последующим удалением фигуры.

### ТЕСТЫ ПРОГРАММЫ:

```
max@max-X550CC:~/oop1$ make
g++ -c main.cpp
g++ main.o Triangle.o Rectangle.o Quadrate.o -o hello
max@max-X550CC:~/oop1$ ./hello
Triangle created: 1, 1, 1
Quadrate created: 1
Rectangle created: 0, 0
Menu
1-cr tr
2-cr quadr
3-cr rect
4-print tr
5-prnt quadr
6-prnt rect
7-prnt sq tr
8-prnt sq quadr
9-prnt sq rect
0-exit
Enter your choise:1
Triangle deleted
4 56 57
Menu
1-cr tr
2-cr quadr
3-cr rect
4-print tr
5-prnt quadr
6-prnt rect
7-prnt sq tr
8-prnt sq quadr
9-prnt sq rect
0-exit
Enter your choise:4
a=4, b=56, c=57
Menu
1-cr tr
2-cr quadr
3-cr rect
```

```
4-print tr
5-prnt quadr
6-prnt rect
7-prnt sq tr
8-prnt sq quadr
9-prnt sq rect
0-exit
Enter your choise:7
109.343
Menu
1-cr tr
2-cr quadr
3-cr rect
4-print tr
5-prnt quadr
6-prnt rect
7-prnt sq tr
8-prnt sq quadr
9-prnt sq rect
0-exit
Enter your choise:2
8
Menu
1-cr tr
2-cr quadr
3-cr rect
4-print tr
5-prnt quadr
6-prnt rect
7-prnt sq tr
8-prnt sq quadr
9-prnt sq rect
0-exit
Enter your choise:8
64
Menu
1-cr tr
2-cr quadr
3-cr rect
4-print tr
5-prnt quadr
6-prnt rect
7-prnt sq tr
8-prnt sq quadr
9-prnt sq rect
0-exit
Enter your choise:9
0
```

Menu

1-cr tr  
2-cr quadr  
3-cr rect  
4-print tr  
5-prnt quadr  
6-prnt rect  
7-prnt sq tr  
8-prnt sq quadr  
9-prnt sq rect  
0-exit

Enter your choise:3

76 2

Menu

1-cr tr  
2-cr quadr  
3-cr rect  
4-print tr  
5-prnt quadr  
6-prnt rect  
7-prnt sq tr  
8-prnt sq quadr  
9-prnt sq rect  
0-exit

Enter your choise:9

152

Menu

1-cr tr  
2-cr quadr  
3-cr rect  
4-print tr  
5-prnt quadr  
6-prnt rect  
7-prnt sq tr  
8-prnt sq quadr  
9-prnt sq rect  
0-exit

Enter your choise:6

a=76, b=2

Menu

1-cr tr  
2-cr quadr  
3-cr rect  
4-print tr  
5-prnt quadr  
6-prnt rect  
7-prnt sq tr  
8-prnt sq quadr

```
9-prnt sq rect
0-exit
Enter your choise:0
Triangle deleted
Quadrate deleted
Rectangle deleted
Exit!
```

## Руководство по использованию программы

При запуске скомпилированного файла запускается диалоговое окно с возможностью выбрать действие:

- 1) Создание восьмиугольника
- 2) Создание пятиугольника
- 3) Создание шестиугольника
- 4) выход (из диалогового меню с последующим завершением программы)

### 3. Листинг программы

#### Figure.h

```
#ifndef FIGURE_H
#define FIGURE_H

class Figure {
public:
    virtual double Square() = 0;
    virtual void Print() = 0;
    virtual ~Figure() {};
};

#endif
```

#### Quadrate.h

```
#ifndef QUADRATE_H
#define QUADRATE_H
#include <cstdlib>
#include <iostream>
#include "Figure.h"

class Quadrate : public Figure{
public:
```

```

    Quadrate();
    Quadrate(std::istream &is);
    Quadrate(size_t i);
    Quadrate(const Quadrate& orig);

    double Square() override;
    void Print() override;

    virtual ~Quadrate();
private:
    size_t side_a;
};

#endif

```

## Quadrate.cpp

```

#include "Quadrate.h"
#include <iostream>
#include <cmath>

Quadrate::Quadrate() : Quadrate(1) {
}

Quadrate::Quadrate(size_t i) : side_a(i){
    std::cout << "Quadrate created: " << side_a << std::endl;
}

Quadrate::Quadrate(std::istream &is) {
    int a;
    is >> a;
    if(a>0){
        side_a=a;
    }
    else{
        std::cout << "Quadrate not created!" << '\n';
    }
}

Quadrate::Quadrate(const Quadrate& orig) {
    std::cout << "Quadrate copy created" << std::endl;
    side_a = orig.side_a;
}

double Quadrate::Square() {
    return (double)(side_a*side_a);
}

void Quadrate::Print() {
    std::cout << "a=" << side_a << std::endl;
}

```

```

Quadrature::~~Quadrature() {
    std::cout << "Quadrature deleted" << std::endl;
}

```

## Rectangle.h

```

#ifndef RECTANGLE_H
#define RECTANGLE_H
#include <cstdlib>
#include <iostream>
#include "Figure.h"

class Rectangle : public Figure{
public:
    Rectangle();
    Rectangle(std::istream &is);
    Rectangle(size_t i, size_t j);
    Rectangle(const Rectangle& orig);

    double Square() override;
    void Print() override;

    virtual ~Rectangle();
private:
    size_t side_a;
    size_t side_b;
};

#endif

```

## Rectangle.cpp

```

#include "Rectangle.h"

#include <iostream>
#include <cmath>

Rectangle::Rectangle() : Rectangle(0, 0) {
}

Rectangle::Rectangle(size_t i, size_t j) : side_a(i), side_b(j) {
    std::cout << "Rectangle created: " << side_a << ", " << side_b
<< std::endl;
}

Rectangle::Rectangle(std::istream &is) {
    int a, b;
    is >> a;
    is >> b;
    if(a>0&&b>0){
        side_a=a;
        side_b=b;
    }
    else{

```



```

        std::cout << "Rectangle not created!" << '\n';
    }
}

Rectangle::Rectangle(const Rectangle& orig) {
    std::cout << "Rectangle copy created" << std::endl;
    side_a = orig.side_a;
    side_b = orig.side_b;
}

double Rectangle::Square() {
    return (double)(side_a*side_b);
}

void Rectangle::Print() {
    std::cout << "a=" << side_a << ", b=" << side_b << std::endl;
}

Rectangle::~Rectangle() {
    std::cout << "Rectangle deleted" << std::endl;
}

```

## Triangle.h

```

#ifndef TRIANGLE_H
#define TRIANGLE_H
#include <cstdlib>
#include <iostream>
#include "Figure.h"

class Triangle : public Figure{
public:
    Triangle();
    Triangle(std::istream &is);
    Triangle(size_t i, size_t j, size_t k);
    Triangle(const Triangle& orig);

    double Square() override;
    void Print() override;

    virtual ~Triangle();
private:
    size_t side_a;
    size_t side_b;
    size_t side_c;
};

#endif

```

## Triangle.cpp

```
#include "Triangle.h"
#include <iostream>
#include <cmath>

int max(int a, int b);
int min(int a, int b);

int max(int a, int b){
    if(a>b){
        return a;
    }
    else{
        return b;
    }
}
int min(int a, int b){
    if(a<b){
        return a;
    }
    else{
        return b;
    }
}

Triangle::Triangle() : Triangle(1, 1, 1) {
}

Triangle::Triangle(size_t i, size_t j, size_t k) : side_a(i),
side_b(j), side_c(k) {
    std::cout << "Triangle created: " << side_a << ", " << side_b <<
", " << side_c << std::endl;
}

Triangle::Triangle(std::istream &is) {
    int a, b, c;
    is >> a;
    is >> b;
    is >> c;
    if(max(a, max(b, c)) >= min(b, c)+min(a, max(b, c))){
        std::cout << "Wrong sides! Triangle not created!" <<
'\n';
    }
    else if(a>0&&b>0&&c>0){
        side_a=a;
        side_b=b;
        side_c=c;
    }
    else{
```

```

        std::cout << "Wrong sizes! Triangle not created!" <<
'\n';
    }
}

Triangle::Triangle(const Triangle& orig) {
    std::cout << "Triangle copy created" << std::endl;
    side_a = orig.side_a;
    side_b = orig.side_b;
    side_c = orig.side_c;
}

double Triangle::Square() {
    double p = double(side_a + side_b + side_c) / 2.0;
    return sqrt(p * (p - double(side_a))*(p - double(side_b))*(p -
double(side_c)));
}

void Triangle::Print() {
    std::cout << "a=" << side_a << ", b=" << side_b << ", c=" <<
side_c << std::endl;
}

```

## main.cpp

```

#include <cstdlib>
#include <iostream>
#include "Triangle.h"
#include "Quadrature.h"
#include "Rectangle.h"
//Бронников Максим №1 по списку М80-204Б-17
//Лабораторная работа №1
/*Необходимо спроектировать и запрограммировать на языке C++ классы
фигур,
согласно вариантов задания.
Классы должны иметь виртуальные метод вывода и метод подсчета площади
Фигуры: Треугольник, Квадрат, Прямоугольник
*/
int main(int argc, char** argv) {
    int i;
    Figure *tr = new Triangle();
    Figure *quadr = new Quadrature();
    Figure *rect = new Rectangle();
    while(1){
        std::cout << "Menu\n1-cr tr\n2-cr quadr\n3-cr rect\n4-print
tr\n5-prnt quadr\n6-prnt rect\n7-prnt sq tr\n8-prnt sq quadr\n9-prnt sq
rect\n0-exit\nEnter your choise:";
        std::cin >> i;
        if(i==1){
            delete tr;
            tr = new Triangle(std::cin);
        }else if(i==2){
            quadr = new Quadrature(std::cin);
        }
    }
}

```

```

}else if(i==3){
    rect = new Rectangle(std::cin);
}else if(i==4){
    tr->Print();
}else if(i==6){
    rect->Print();
}else if(i==5){
    quadr->Print();
}else if(i==7){
    std::cout << tr->Square() << std::endl;
}else if(i==8){
    std::cout << quadr->Square() << std::endl;
}else if(i==9){
    std::cout << rect->Square() << std::endl;
}else if(i==0){
    delete tr;
    delete quadr;
    delete rect;
    std::cout << "Exit!" << '\n';
    return 0;
}
}
}

```

## Вывод:

В ходе выполнения данной лабораторной я научилась работать с классами, разобралась, что такое конструктор и деструктор, повторила материал о создании диалогового меню. Самым интересным для меня оказался поиск необходимых параметров для задания фигур.

## СПИСОК ЛИТЕРАТУРЫ

1. Справочник по языку с и с++ [Электронный ресурс]. URL : <http://www.c-cpp.ru> (дата обращения : 28.09.2018)
2. Все формулы площади трапеции [Электронный ресурс]. URL : <http://100formul.ru/48> (дата обращения : 28.09.2018)
3. Видеоуроки по программированию на с++ [Электронный ресурс]. URL : <https://www.youtube.com/watch?v=kRcbYlK3OnQ&list=PLQOaTSbfxUtCrKs0nicOg2npJQYSPGO9r> (дата обращения : 28.09.2018)