Московский Авиационный Институт

(Национальный Исследовательский Университет)

Кафедра: 806 «Вычислительная математика и программирование»

Факультет: «Прикладная математика и физика»

Дисциплина: «Объектно-ориентированное программирование»

Лабораторная работа №1.

Тема:

«Классы»

Группа: 8О-207Б

Студент: Сорокин Денис Михайлович

Преподаватель: Поповкин Александр Викторович

Вариант: №33

Москва

2016

# Цель работы

* Программирование классов на языке С++
* Управление памятью в языке С++
* 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.
* Должный иметь общий виртуальный метод расчета площади фигуры – Square.
* Должны иметь конструктор, считывающий значения основных параметров фигуры из стандартного потока cin.
* Должны быть расположены в раздельных 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.

# Листинг кода

## figure.h

#ifndef FIGURE\_H

#define FIGURE\_H

class Figure {

public:

virtual double Square() = 0;

virtual void Print() = 0;

virtual ~Figure() {};

};

#endif

## quadro.h

#ifndef QUADRO\_H

#define QUADRO\_H

#include <cstdlib>

#include <iostream>

#include "figure.h"

class Quadro : public Figure {

public:

Quadro(std::istream &is);

double Square() override;

void Print() override;

virtual ~Quadro();

private:

size\_t side;

};

#endif

## quadro.cpp

#include "quadro.h"

#include <iostream>

Quadro::Quadro(std::istream &is) {

std::cout << "Insert side of quadro: ";

is >> side;

}

double Quadro::Square() {

return (double)side \* side;

}

void Quadro::Print() {

std::cout << "Side = " << side << std::endl;

}

Quadro::~Quadro() {

std::cout << "Quadro deleted" << std::endl;

}

## rectangle.h

#ifndef RECTANGLE\_H

#define RECTANGLE\_H

#include <cstdlib>

#include <iostream>

#include "figure.h"

class Rectangle : public Figure {

public:

Rectangle(std::istream &is);

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>

Rectangle::Rectangle(std::istream &is) {

std::cout << "Insert side A: ";

is >> side\_a;

std::cout << "Insert side B: ";

is >> side\_b;

}

double Rectangle::Square() {

return (double)side\_a \* side\_b;

}

void Rectangle::Print() {

std::cout << "a = " << side\_a << std::endl

<< "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);

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>

Triangle::Triangle(std::istream &is) {

std::cout << "Insert side A: ";

is >> side\_a;

std::cout << "Insert side B: ";

is >> side\_b;

std::cout << "Insert side C: ";

is >> 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 << std::endl

<< "b = " << side\_b << std::endl

<< "c = " << side\_c << std::endl;

}

Triangle::~Triangle() {

std::cout << "Triangle deleted" << std::endl;

}

## main.cpp

#include <cstdlib>

#include "triangle.h"

#include "quadro.h"

#include "rectangle.h"

void Tips() {

std::cout << "1 - Triangle" << std::endl

<< "2 - Quadro" << std::endl

<< "3 - Rectangle" << std::endl

<< "0 - end" << std::endl;

}

int main(int argc, char\*\* argv) {

Tips();

while (1) {

short i = 0;

std::cin >> i;

Figure \*ptr = NULL;

if (i == 0) {

break;

}

if (i == 1) {

ptr = new Triangle(std::cin);

}

if (i == 2) {

ptr = new Quadro(std::cin);

}

if (i == 3) {

ptr = new Rectangle(std::cin);

}

if (ptr == NULL) {

std::cout << "Wrong input, try again" << std::endl;

continue;

}

ptr->Print();

std::cout << "Square = " << ptr->Square() << std::endl;

delete ptr;

}

return 0;

}

# Выводы