МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Белгородский государственный технологический университет

им. В.Г. Шухова»

ИНСТИТУТ ЭНЕРГЕТИКИ, ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

И УПРАВЛЯЮЩИХ СИСТЕМ

ЛАБОРАТОРНАЯ РАБОТА №4

Дисциплина: Объектно-ориентированное программирование

Тема: «Классы»

Работу выполнил:

Ст. группы ПВ-223

Дмитриев А.А.

Проверил:

Черников С.В.

Белгород 2024

**Цель работы:** приобретение практических навыков создания класса на языке С++.

**Задание**: Выполнить построение диаграммы объектов в соответствии с заданием варианта (табл. 1). По заданному описанию класса в соответствии с вариантом создать класс на языке С++ и переопределить указанные операции.

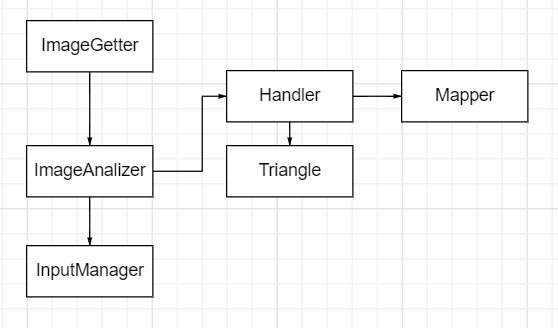
**Вариант 2**

|  |  |
| --- | --- |
| №1  Выполнить построение объектной модели следующей предметной области: “система триангуляции изображений” | №2  Создать класс Triangle, разработав следующие элементы класса:  а. Поля: int a, b, c;  б. Конструктор, позволяющий создать экземпляр класса с заданными длинами сторон.  в. Методы, позволяющие:  вывести длины сторон треугольника на экран;  рассчитать периметр треугольника;  рассчитать площадь треугольника.  г. Перегрузить (переопределить):  сравнение (больше);  сложение (унарный, префиксное). |

**Задание №1**

***«Система триангуляции изображений»***

Процесс разбиения полигональной области со сложной конфигурацией в набор треугольников называется ***триангуляцией.***

******

**Задание №2**

**Main.cpp**

#include <iostream>

#include <locale>

#include "Triangle.h"

int main()

{

    setlocale(LC\_ALL, "Rus");

    int a = 0,

        b = 0,

        c = 0;

    while (true)

    {

        cout << " Input sides of Triangle a, b, c = ";

        cin >> a;

        cin >> b;

        cin >> c;

        Triangle T\_sides = Triangle(a, b, c);

        if (T\_sides.existTriangle())

        {

            cout << T\_sides.AreaTriangle() << endl;

            cout << T\_sides.PerimTriangle() << endl;

break;

        }

        else

            cout << " Triangle doesn't exist!\n Input other values!\n\n";

    }

    //больше

    if (a > b)

        cout << "a > b = True" << endl;

    else

        cout << "a > b = False" << endl;

    //унарное сложение

    cout << "a + 3 = " << a + 3 << endl;

    //префиксный инкремент

    b++;

    cout << "b++   = " << b << endl;

    system("pause");

}

**Модуль Triangle.h**

#ifndef OOP\_LABA\_4\_TRIANGLE\_H

#define OOP\_LABA\_4\_TRIANGLE\_H

#include <iostream>

using namespace std;

//класс треугольник

class Triangle

{

private:

    int a, b, c;       //поля

public:

    //Конструктор, позволяющий создать экземпляр класса

    //с заданными длинами сторон треугольника

    Triangle(int a1, int b1, int c1)

    {

        a = a1;

        b = b1;

        c = c1;

    }

    //Вывод длин сторон треугольника в консоль

    Triangle& printSides();

    //Возвращает периметр треугольника

    int PerimTriangle();

    //Возвращает площадь треугольника

    double AreaTriangle();

    //Проверка на существование треугольника

    bool existTriangle();

    Triangle (int i) : a(i), b(i), c(i) {}

    //унарный +

    friend const Triangle& operator+(const Triangle&);

    //префиксный инкремент ++

    friend const Triangle& operator++(Triangle&);

    //сравнение больше >

    friend bool operator> (Triangle&, Triangle&);

};

#endif //OOP\_LABA\_4\_TRIANGLE\_H

**Модуль Triangle.cpp**

#include "Triangle.h"

//Вывод длин сторон треугольника в консоль

Triangle& Triangle::printSides()

{

    cout << "Triangle sides: " << endl;

    cout << "a = " << a << endl;

    cout << "b = " << b << endl;

    cout << "c = " << c << endl;

    return \*this;

}

//Рассчитывает периметр треугольника

int Triangle::PerimTriangle()

{

    cout << "Triangle perimeter: ";

    return  a + b + c;

}

//Рассчитывает площадь треугольника

double Triangle::AreaTriangle()

{

    cout << "Triangle square: ";

    double p = (a + b + c) / 2;

    return (sqrt(p \* (p - a) \* (p - b) \* (p - c)));

}

//Проверка на существование треугольника

bool Triangle::existTriangle()

{

    return a < b + c && b < a + c && c < b + a;

}

//унарный плюс ничего не делает

const Triangle& operator+(const Triangle& i) {

    return i.a;

}

//префиксная версия возвращает значение после инкремента

const Triangle& operator++(Triangle& i) {

    i.b++;

    return i;

}

//оператор больше

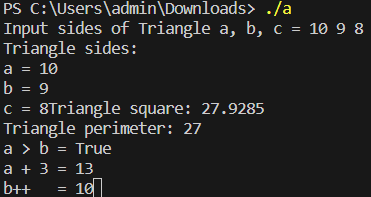
bool operator>(Triangle& a, Triangle& b)

{

   return a > b;

}

**Тестовые данные**



**Вывод:** В ходе лабораторной работы приобрели практические навыки создания классов на языке С++.