Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«**Пермский национальный исследовательский политехнический университет»**

Кафедра «Информационные технологии и автоматизированные системы»

**ОТЧЕТ**

Дисциплина: «Информатика»

"Классы и объекты.Принцип подстановки"

Семестр 2

Выполнил работу

Студент группы РИС-22-1Б

Бадртдинов Т.З

Проверил

Доцент кафедры ИТАС

Полякова О.А.

Г. Пермь-2023

# Постановка задачи

1. Определить пользовательский класс.

2. Определить в классе следующие конструкторы: без параметров, с параметрами, копирования.

3. Определить в классе деструктор.

4. Определить в классе компоненты-функции для просмотра и установки полей данных (селекторы и модификаторы).

5. Перегрузить операцию присваивания.

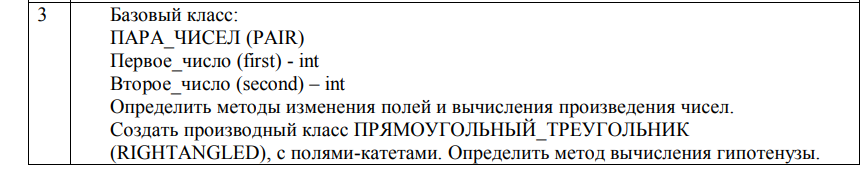
6. Перегрузить операции ввода и вывода объектов с помощью потоков.

7. Определить производный класс.

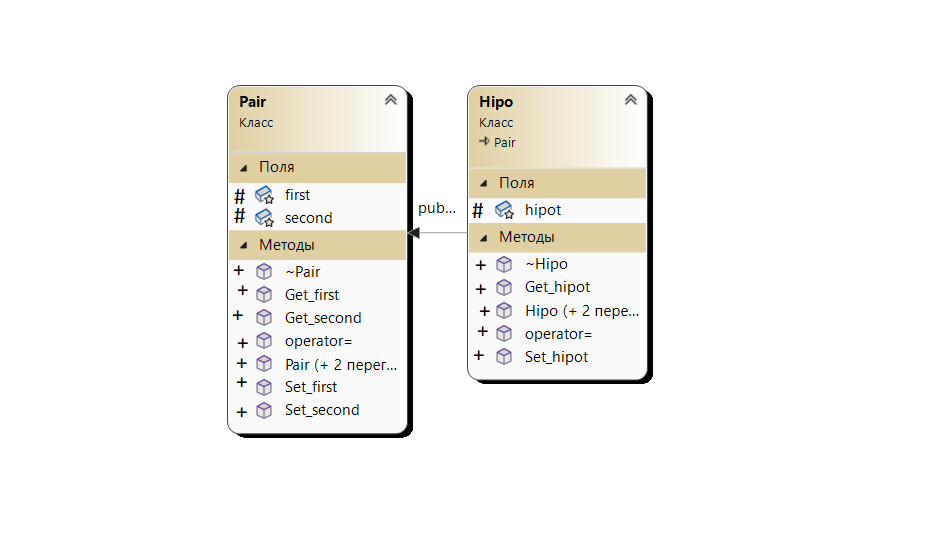
8. Написать программу, в которой продемонстрировать создание объектов и работу всех перегруженных операций.

9. Реализовать функции, получающие и возвращающие объект базового класса. Продемонстрировать принцип подстановки.

Вариант 3



Описание класса

**

# Код программы(hipotenuse.cpp)

#include "Hipotenuse.h"

#include <math.h>

//конструктор без параметров

Hipo::Hipo(void):Pair()

{

hipot=0;

}

//дестрктор

Hipo::~Hipo(void) { }

//конструктор с параметрами

Hipo::Hipo(int C, int P, int G):Pair(C,P)

{

hipot= sqrt(C \* C+ P \* P);

}

//конструктор копирования

Hipo::Hipo(const Hipo &L)

{

first=L.first;

second=L.second;

}

//модификатор

void Hipo::Set\_hipot(int G)

{

hipot=G;

}

//оперция присваивания

Hipo& Hipo::operator=(const Hipo& l)

{

if (&l == this)return \*this;

second = l.second;

first = l.first;

hipot = l.hipot;

return \*this;

}

//операция ввода

istream& operator>>(istream&in,Hipo&l)

{

cout<<"\nfirst:";in>>l.first;

cout<<"\nsecond:";in>>l.second;

l.hipot = sqrt(l.first \* l.first + l.second \* l.second);

return in;

}

//операция вывода

ostream& operator<<(ostream& out, const Hipo& l)

{

out << "\nfirst : " << l.first;

out << "\nsecond : " << l.second;

out << "\nhipot : " << l.hipot;

out << "\n";

return out;

}

# Код программы(pair.cpp)

#include "Pair.h"

//конструктор без параметров

Pair::Pair(void)

{

; first=0; second=0;

} //деструктор

Pair::~Pair(void) { }

//конструктор с параметрами

Pair::Pair(int C,int P)

{

first=C; second=P;

} //конструктор копирования

Pair::Pair(const Pair& t)

{

first = t.first;

second = t.second;

}

//модификаторы

void Pair::Set\_first(int C)

{

first=C;

}

void Pair::Set\_second(int P)

{

second=P;

}

//перегрузка операции присваивания

Pair& Pair::operator=(const Pair&c)

{

if(&c==this)return \*this;

second=c.second;

first=c.first;

return \*this;

}

//глобальная функция для ввода

istream& operator>>(istream&in,Pair&c)

{

cout<<"\nfirst:";in>>c.first;

cout<<"\nsecond:";in>>c.second;

return in;

}

//глобальная функция для вывода

ostream& operator<<(ostream&out,const Pair&c)

{

out<<"\nfirst: "<<c.first;

out<<"\nsecond : "<<c.second;

out<<"\n";

return out;

}

# Код программы(main)

#include <iostream>

#include "Pair.h"

#include "Hipotenuse.h"

using namespace std;

void main() {

Pair a;

cin>>a;

cout<<a;

Hipo c;//создать объект

cin>>c;//ввести значения атрибутов

cout<<c;//вывести значения атрибутов

# }

# Работы программы

