ФАКУЛЬТЕТ «Информатика и системы управления»  
КАФЕДРА «Информационная безопасность»

ОТЧЕТ  
по лабораторной работе № 3  
по учебной дисциплине «Алгоритмические языки»  
на тему: «Изучение возможностей наследования классов»

Вариант 22

Выполнила:   
Студентка 1 курса, гр. ИУ8-24  
Степаненко Александра

**Цель работы:**

Овладение навыками разработки программ на языке Си++, использующих возможности наследования классов для решения различных задач.

**Задачи работы:**

1. Изучить необходимые учебные материалы, посвященные наследованию классов в языке Си++
2. Разработать программу на языке Си++ для решения заданного варианта задания
3. Отладить программу
4. Представить результаты работы программы
5. Подготовить отчет по лабораторной работе

**Условие задачи:**

Создать базовый класс «вектор на плоскости». Элементы класса: поля, задающие координаты точки (статус доступа *protected*), определяющей конец вектора (начало вектора находится в точке с координатами 0, 0); конструктор для инициализации полей; функция для печати координат вектора. Создать производный класс «вектор в трехмерном пространстве». Элементы класса: дополнительное поле, задающее дополнительную координату; конструктор для инициализации полей; переопределенная функция для печати координат вектора (внутри переопределенной функции должна вызываться функция из базового класса). Создать по 1 объекту каждого из классов. Показать вызов созданных функций. При переопределении функций обеспечить и продемонстрировать два варианта: статический полиморфизм и динамический полиморфизм.

**Выполнение работы:**

#include <iostream>

#include <cmath>

using namespace std;

class VectorOnThePlane {

protected:

double x, y;

public:

VectorOnThePlane(double x, double y) {

this->x = x;

this->y = y;

}

virtual void print() {

cout << "x= " << x << " y= " << y << " ";

}

};

class VectorInThreeSpace : public VectorOnThePlane {

double z;

public:

VectorInThreeSpace(double x, double y, double z) : VectorOnThePlane(x, y) {

this->z = z;

}

void print() {

VectorOnThePlane::print();

cout << " z= " << z << endl;

}

};

int main()

{

VectorOnThePlane vector1(2, 3);

VectorInThreeSpace vector2(1, 2, 3);

vector1.print();

vector2.print();

VectorOnThePlane\* vector1Ptr = new VectorOnThePlane(4, 6);

vector1Ptr->print();

VectorInThreeSpace\* vector2Ptr = new VectorInThreeSpace(7, 8, 9);

vector2Ptr->print();

return 0;

}

**Вывод:**

В результате выполнения лабораторной работы мы успешно создали базовый класс «вектор на плоскости», производный класс «вектор в трехмерном пространстве», продемонстрировали использование статического и динамического полиморфизма.