|  |  |
| --- | --- |
|  | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет  имени Н. Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н. Э. Баумана)** |

|  |  |
| --- | --- |
| ФАКУЛЬТЕТ | «Информатика, искусственный интеллект и системы управления» (ИУ) |

|  |  |
| --- | --- |
| КАФЕДРА | «Информационная безопасность» (ИУ8) |

Лабораторная работа № 3

ПО КУРСУ

«Алгоритмические языки»

на тему «Изучение перегрузки стандартных операций в языке Си++»

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Студент | ИУ8-24 |  |  |  | Соболева Д.Е. |
|  | (Группа) |  |  |  | (И. О. Фамилия) |
|  |  |  |  |  |  |
| Преподаватель: |  |  |  |  | Барыкин Д.В.  Быков А.Ю. |
|  |  |  |  |  | (И.О. Фамилия) |

**Цель работы** состоит в овладении навыками разработки программ на языке Си++, использующих возможности наследования классов для решения различных задач.

**Вариант 22**

**Задача**

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.

**Код программы:**

#include <iostream>

using namespace std;

// Базовый класс "вектор на плоскости"

class Vector2D {

protected:

double x;

double y;

public:

Vector2D(double x\_val, double y\_val) : x(x\_val), y(y\_val) {}

void printCoordinates() {

cout << "2D Vector coordinaty: (" << x << ", " << y << ")" << endl;

}

};

// Производный класс "вектор в трехмерном пространстве"

class Vector3D : public Vector2D {

private:

double z;

public:

Vector3D(double x\_val, double y\_val, double z\_val) : Vector2D(x\_val, y\_val), z(z\_val) {}

void printCoordinates() {

cout << "3D Vector koordinaty: (" << x << ", " << y << ", " << z << ")" << endl;

}

void printCoordinatesDynamic() {

Vector2D::printCoordinates();

cout << "Dobavili coordinatu: " << z << endl;

}

};

int main() {

// Создание объектов

Vector2D vec2D(1.0, 2.0);

Vector3D vec3D(3.0, 4.0, 5.0);

// Вызов функций для базового класса и производного класса

vec2D.printCoordinates();

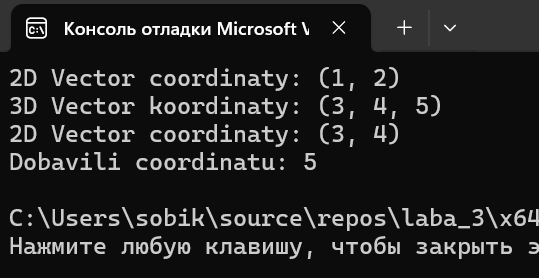
vec3D.printCoordinates();

vec3D.printCoordinatesDynamic();

return 0;

}

Консоль:



**Контрольный расчёт:** Пересчитав руками данные параметры, я получила точно такие же ответы.

**Вывод:** Выполнив лабораторную работу №3, я изучила тему «Изучение возможностей наследования классов в языке Си++» и применила классы и их наследование на практике.