ПРИЛОЖЕНИЕ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТЕЛЕКОММУНИКАЦИЙ ИМ. ПРОФ. М.А. БОНЧ-БРУЕВИЧА»

(СПбГУТ)

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Кафедра Систем обработки данных

**Дисциплина «Технологии программирования»**

**ОТЧЕТ**

**за практическое занятие №4**

**Тема: Взаимодействие функций в программах**

Выполнил

Студент 2 курса, гр. ИБ-32вп

\_\_\_\_\_\_\_\_\_\_\_\_\_ Братишкин Д.Е.

Принял

Доцент кафедры БИС

\_\_\_\_\_\_\_\_\_\_\_\_\_ Медведев В.А.

Санкт-Петербург

2024

Тема: **Проектирование классов и создание объектов**

Цель: Получение практических навыков разработки классов, создания объектов и использования дружественных функций для обработки данных.

Задание на практическое занятие

Вариант С.

Задание 1.

- Спроектировать класс `Point`, обозначающий координаты точки на плоскости: `x`, `y`.

- а) Разработать конструктор с параметрами для инициализации объектов класса `Point`.

- б) Разработать функции доступа к закрытым членам класса.

- в) Разработать деструктор, обозначающий момент уничтожения объекта.

- Создать два объекта, соответствующих двум точкам, и определить расстояние между ними.

Задание 2.

- Спроектировать класс в соответствии с индивидуальным заданием (см. таблица 6).

- Создать несколько объектов.

- Рассчитать вычисляемый показатель.

Текст программы с комментариями

#include <iostream>

#include <cmath>

using namespace std;

// Задание 1

class Point {

private:

    float x, y; // координаты точки

public:

    // Конструктор для инициализации координат

    Point(float xCoord, float yCoord) : x(xCoord), y(yCoord) {}

    // Метод для получения координаты X

    float getX() { return x; }

    // Метод для получения координаты Y

    float getY() { return y; }

    // Деструктор, который сообщает о разрушении объекта

    ~Point() {

        cout << "Object at (" << x << ", " << y << ") is destroyed." << endl;

    }

    // Статический метод для вычисления расстояния между двумя точками

    static float distance(Point p1, Point p2) {

        return sqrt(pow(p2.x - p1.x, 2) + pow(p2.y - p1.y, 2));

    }

};

// Задание 2

class LocalNetwork {

private:

    int numberOfDevices; // количество устройств в сети

    float installationCostPerDevice; // стоимость установки на одно устройство

public:

    // Конструктор для инициализации количества устройств и стоимости установки

    LocalNetwork(int devices, float cost) : numberOfDevices(devices), installationCostPerDevice(cost) {}

    // Метод для расчета минимальной стоимости установки

    float calculateMinimumInstallationCost() {

        return numberOfDevices \* installationCostPerDevice;

    }

    // Деструктор, который сообщает о разрушении объекта

    ~LocalNetwork() {

        cout << "LocalNetwork object destroyed." << endl;

    }

};

int main() {

    // Создание двух точек с заданными координатами

    Point point1(1.0, 2.0);

    Point point2(4.0, 6.0);

    // Вычисление расстояния между двумя точками

    float distanceBetweenPoints = Point::distance(point1, point2);

    cout << "Distance between points: " << distanceBetweenPoints << endl;

    // Создание локальной сети с 10 устройствами и стоимостью установки 150.0

    LocalNetwork network(10, 150.0);

    // Расчет минимальной стоимости установки

    float installationCost = network.calculateMinimumInstallationCost();

    cout << "Minimum installation cost: " << installationCost << endl;

    return 0; // Завершение программы

}

Результаты работы программы

Задание 1

- Ввод: Создание объектов `point1` и `point2` с координатами (1.0, 2.0) и (4.0, 6.0) соответственно.

- Вывод: Расстояние между точками: 5.0

Задание 2

- Ввод: Создание объекта `network` с 10 устройствами и стоимостью монтажа 150.0 за устройство.

- Вывод: Минимальная стоимость монтажа: 1500.0

Выводы по разработанным элементам программы

Программа успешно выполняет все поставленные задачи. В первом задании она корректно создает объекты класса `Point`, инициализирует их координаты, вычисляет расстояние между точками и выводит результат. Во втором задании программа создает объекты класса `LocalNetwork`, вычисляет минимальную стоимость монтажа и выводит результат.

Программа оформлена в виде классов с конструкторами, функциями доступа и деструкторами, что делает код более структурированным и легким для понимания и отладки. Обработка ошибок и ввод данных пользователем выполнены корректно, что повышает надежность и удобство использования программы.