Міністерство освіти та науки України

Інститут «Телекомунікації, радіоелектроніки та радіотехніки»

Изображение выглядит как текст, круг, Шрифт, эмблема

Автоматически созданное описание

Звіт

З лабораторної роботи №5

З дисципліни «Об`єктно-орієнтоване програмування»

На тему: «Робота з класами з різними конструкторами»

Виконав:

Студент групи АП-22

Іщак Д.А.

Прийняв:

Алтунін С. І

Львів 2024

**Мета:** Створити, відлагодити та протестувати програму, у якій створити клас для обробки даних.

**Хід роботи:**

1. продумайте, які типи змінних вам потрібно використати для виконання завдання.

2. при створенні екземпляру класу дайте йому ім’я відповідно до призначення класу, наприклад Student John.

3. реалізуйте конструктор за замовчуванням, конструктори з параметрами, деструктор.

4. використайте конструктор з параметрами для збереження даних.

5. реалізуйте методи класу для введення і виведення даних.

6. реалізуйте кілька екземплярів заданого класу.

7. методи зміни полів класу повинні перевіряти валідність заданих змінних і попереджувати користувача про некоректні значення.

8. додайте можливість виведення інформації про екземпляр класу за певним критерієм, наприклад всіх студентів 2 курсу, всіх студентів із середнім балом вище 4, всіх службовців відповідної посади чи старших/молодших N років, всіх жителів вул. Шевченка, і т.д. Якщо інформація за певним критерієм відсутня, то потрібно вивести відповідне повідомлення.

9. можете збільшити кількість членів класу додатковими полями чи методами, наприклад для копіювання даних із вашого класу.

10.програма має вивести на консоль (термінал) вхідні дані та результат виконання.

**Варіант завдання:**

****

**Код програми:**

**#include <iostream>**

**#include <vector>**

**#include <string>**

**class Car {**

**private:**

**std::string brand;**

**std::string model;**

**int year;**

**public:**

**// Конструктор за замовчуванням**

**Car() : brand(""), model(""), year(0) {}**

**// Конструктор з параметрами**

**Car(std::string b, std::string m, int y) : brand(b), model(m), year(y) {**

**if (y < 1886 || y > 2024) {**

**std::cerr << "Некоректний рік випуску! Рік встановлено на 0.\n";**

**year = 0;**

**}**

**}**

**// Методи введення даних**

**void setBrand(const std::string& b) { brand = b; }**

**void setModel(const std::string& m) { model = m; }**

**void setYear(int y) {**

**if (y >= 1886 && y <= 2024) {**

**year = y;**

**}**

**else {**

**std::cerr << "Некоректний рік випуску!\n";**

**}**

**}**

**// Методи виведення інформації**

**void displayInfo() const {**

**std::cout << "Марка: " << brand << ", Модель: " << model**

**<< ", Рік випуску: " << year << "\n";**

**}**

**// Геттери**

**int getYear() const { return year; }**

**};**

**// Функція пошуку автомобілів за критерієм**

**void findCarsByYear(const std::vector<Car>& cars, int year) {**

**bool found = false;**

**for (const auto& car : cars) {**

**if (car.getYear() == year) {**

**car.displayInfo();**

**found = true;**

**}**

**}**

**if (!found) {**

**std::cout << "Автомобілів " << year << " року немає.\n";**

**}**

**}**

**int main() {**

**// Створення екземплярів класу**

**Car car1("Toyota", "Camry", 2020);**

**Car car2("BMW", "X5", 2018);**

**Car car3("Ford", "Focus", 2020);**

**// Збереження у вектор**

**std::vector<Car> cars = { car1, car2, car3 };**

**// Виведення інформації про всі автомобілі**

**std::cout << "Список автомобілів:\n";**

**for (const auto& car : cars) {**

**car.displayInfo();**

**}**

**// Пошук автомобілів за роком**

**int searchYear = 2020;**

**std::cout << "\nПошук автомобілів " << searchYear << " року:\n";**

**findCarsByYear(cars, searchYear);**

**return 0;**

**}**

**Результати виконання роботи:**

**Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание**

**Блок-схема програми:**

**Изображение выглядит как текст, чек, снимок экрана, документ

Автоматически созданное описание**

Висновок:

У цій лабораторній роботі реалізовано клас "Вантажівка" із конструкторами за замовчуванням і з параметрами, а також методами для перевірки коректності введених даних. Програма дозволяє ефективно створювати та обробляти об’єкти класу, забезпечуючи зручність у роботі з даними про вантажівки. Реалізація перевірки валідності даних допомагає уникнути помилок, спричинених некоректними значеннями.