Міністерство освіти та науки України

Інститут «Телекомунікації, радіоелектроніки та радіотехніки»

Изображение выглядит как текст, круг, Шрифт, эмблема

Автоматически созданное описание

Звіт

З лабораторної роботи №6

З дисципліни «Об`єктно-орієнтоване програмування»

На тему: «Робота з статичними масивами»

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. Одним з членів класу є статичний масив структур, в якому зберігаються введені дані.

**Хід роботи:**

1. продумайте, які типи змінних вам потрібно використати для виконання завдання.
2. створіть проєкт із кількома файлами, тобто крім основного файлу, наприклад main.cpp, додайте два файл, наприклад student.h та student.cpp. Для цього потрібно виправити файл ***..\.vscode\tasks.json***:

{

"version": "2.0.0", "tasks": [

{

"type": "cppbuild",

"label": "C/C++: g++.exe build active file", "command": "C:\\msys64\\ucrt64\\bin\\g++.exe", "args": [

"-fdiagnostics-color=always", "-g",

| замість відносного шляху до виконуваного файла: | вказати відносний шлях директорії проєкту до всіх файлів \*.cpp, /\*.h: |
| --- | --- |
| "${file}", | "${workspaceFolder}/\*.cpp", "${workspaceFolder}/\*.h", |

"-o",

"${fileDirname}\\${fileBasenameNoExtension}.exe"

],

"options": {

"cwd": "${fileDirname}"

},

"problemMatcher": [ "$gcc"

],

"group": {

"kind": "build", "isDefault": true

},

"detail": "compiler: C:\\msys64\\ucrt64\\bin\\g++.exe"

}

]

}

1. У заголовному файлі \*.h оголосіть клас:

class Student{

private

std::array<student\_s\_t, SIZE> students;

…

}

1. оголосіть відповідну структуру, наприклад student\_s\_t,перед

оголошенням класу:

struct student\_s\_t{ int id; std::string name; int course;

};

1. реалізуйте для класу конструктор та деструктор.
2. реалізуйте методи класу для введення і виведення даних.
3. методи зміни полів класу повинні перевіряти валідність заданих змінних і попереджувати користувача про некоректні значення.
4. можете збільшити кількість членів класу додатковими полями чи методами.
5. програма має вивести на консоль (термінал) вхідні дані та результат виконання.

**Варіант завдання:**

 

**Код програми:**

**main.cpp:**

#include <iostream>

#include <string>

#include <limits>

// Базовий клас Car

class Car {

protected:

std::string brand;

std::string model;

public:

// Конструктор базового класу

Car(const std::string& brand = "", const std::string& model = "")

: brand(brand), model(model) {}

// Методи для вводу даних

void input() {

std::cout << "Введіть марку автомобіля: ";

std::cin >> brand;

std::cout << "Введіть модель автомобіля: ";

std::cin >> model;

}

// Метод для виводу даних

void display() const {

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

}

};

// Дочірній клас CarDetails

class CarDetails : public Car {

private:

int year;

public:

// Конструктор дочірнього класу

CarDetails(const std::string& brand = "", const std::string& model = "", int year = 0)

: Car(brand, model), year(year) {}

// Метод для вводу даних

void input() {

Car::input(); // Виклик методу вводу базового класу

while (true) {

std::cout << "Введіть рік випуску автомобіля: ";

std::cin >> year;

// Валідація року

if (std::cin.fail() || year < 1886 || year > 2024) {

std::cout << "Некоректний рік. Спробуйте ще раз.\n";

std::cin.clear();

std::cin.ignore(std::numeric\_limits<std::streamsize>::max(), '\n');

}

else {

break;

}

}

}

// Метод для виводу даних

void display() const {

Car::display(); // Виклик методу виводу базового класу

std::cout << ", Рік випуску: " << year << '\n';

}

};

int main() {

// Створення об'єкта дочірнього класу

CarDetails car;

std::cout << "Введіть дані про автомобіль:\n";

car.input(); // Введення даних користувачем

std::cout << "\nВведені дані автомобіля:\n";

car.display(); // Вивід даних на екран

return 0;

}

**Результати виконання роботи:**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, Шрифт

Автоматически созданное описание**

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

**Блок-схема програми:**

**Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Автоматически созданное описание**

Висновок:

У ході лабораторної роботи було створено програму для роботи зі статичними масивами за допомогою класу. Було реалізовано структуру для зберігання даних про вантажівки та відповідний клас із методами для додавання, перевірки валідності введених даних та їх виведення. Програма успішно перевіряє введені дані, обробляє їх та повідомляє користувача про помилки, якщо вони виникають. Вдалося забезпечити зручність роботи через використання об’єктно-орієнтованого підходу. Реалізовані методи гарантують коректність і стабільність роботи програми, а також дозволяють ефективно управляти даними в статичному масиві.