**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**ВСП – Технічний коледж**

**НУ «Львівська політехніка»**

Відділення

Інформаційних технологій

Та комп`ютерної техніки

**ЗВІТ**

Про виконання лабораторної роботи №5

З предмету «Навчальна практика з основ об'єктно-орієнтованого програмування»

Виконав

Студент групи 31 ПЗ

Тисляк А.О.

Перевірив викладач:

Чистяк В.В.

Львів 2021

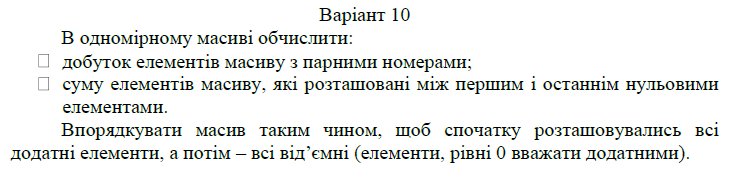
**Лабораторна робота №5 «Наслідування в С++. Віртуальні функції. Чисто віртуальні функції та абстрактні класи»**

**Мета роботи:** ознайомитись з поняттям та призначенням шаблонних функцій та шаблонних класів. Навчитися використовувати шаблони класів та шаблони функцій в мові С++.

*Варіант 10*

**Завдання:** абстрактний базовий клас містить чисто віртуальні методи для отримання даних ії додавання, редагування та видалення. Два дочірні класи містять реалізацію чисто віртуальних функції для введення/виведення даних на консоль (перший клас) та у файл (другий клас).

**Завдання мого варіанту:**



**Код програми:**

***Classes.h***

#pragma once

#include <iostream>

#include <fstream>

#include <istream>

#include <ostream>

#include <vector>

#include <algorithm>

template <class T>

class BaseClass {

public:

std::vector<T> vec;

BaseClass() = default;

virtual ~BaseClass() = default;

virtual void get\_data(std::istream\*) = 0;

virtual void print\_data(std::ostream\*) = 0;

T even\_num\_prod() const {

T prod = 1;

for (int i = 1; i < this->vec.size(); i += 2)

prod \*= this->vec[i];

return prod;

}

T sum\_betw\_zeros() const {

T sum = 0;

std::vector<int> toFind = { 0 };

auto a = std::find(this->vec.begin(), this->vec.end(), 0);

auto b = std::find\_end(this->vec.begin(), this->vec.end(), toFind.begin(), toFind.end());

for (auto i = a; i != b; i++)

sum += \*i;

return sum;

}

void pos\_neg\_sort() {

std::sort(this->vec.begin(), this->vec.end(), [](const T& a, const T& b) {

if (a < 0 && b >= 0) return false;

if (a >= 0 && b < 0) return true;

if (a < 0 && b < 0) return a > b;

return a < b;

});

}

};

template <class T>

class Child1 : public BaseClass<T> {

public:

Child1() = default;

virtual ~Child1() = default;

virtual void get\_data(std::istream\* is) override {

int size;

std::cout << "Enter the array`s size: ";

\*is >> size;

this->vec.resize(size);

std::cout << "\nEnter your array:\n";

for (int i = 0; i < size; i++)

\*is >> this->vec[i];

}

virtual void print\_data(std::ostream\* os) override {

int j = 0;

for (auto i = this->vec.begin(); i != this->vec.end(); i++, j++)

\*os << "arr[" << j << "] = " << \*i << "\n";

}

};

template <class T>

class Child2 : public BaseClass<T> {

public:

Child2() = default;

virtual ~Child2() = default;

virtual void get\_data(std::istream\* ifs) override {

int size;

\*ifs >> size;

this->vec.resize(size);

for (int i = 0; i < size; i++)

\*ifs >> this->vec[i];

}

virtual void print\_data(std::ostream\* ofs) override {

\*ofs << "\nProd = " << this->even\_num\_prod() << "\n";

\*ofs << "Sum = " << this->sum\_betw\_zeros() << "\n\n";

this->pos\_neg\_sort();

int j = 0;

for (auto i = this->vec.begin(); i != this->vec.end(); i++, j++)

\*ofs << "arr[" << j << "] = " << \*i << "\n";

}

};

***Main5.cpp***

#include <iostream>

#include <Windows.h>

#include <fstream>

#include "Classes.h"

int main() {

using namespace std;

ifstream inf("file1.txt");

ofstream outf("file2.txt");

Child1<int> arr;

arr.get\_data(&std::cin);

cout << "\nProd = " << arr.even\_num\_prod() << "\n";

cout << "Sum = " << arr.sum\_betw\_zeros() << "\n\n";

arr.pos\_neg\_sort();

cout << "Sorted:\n";

arr.print\_data(&std::cout);

Child2<int> arr2;

arr2.get\_data(&inf);

arr2.print\_data(&outf);

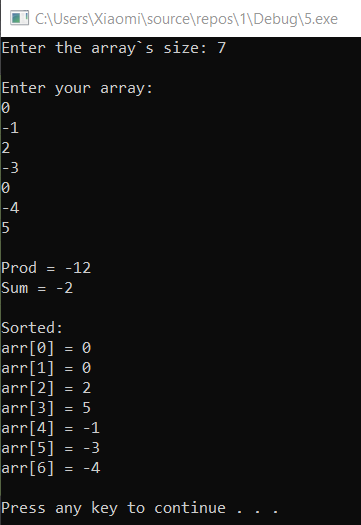
cout << "\n";

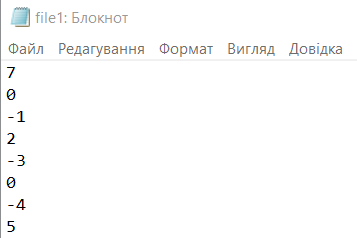
system("pause");

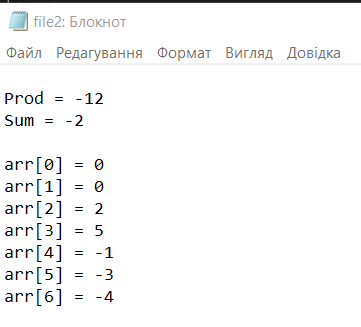
return 0;

}

***Скрін-шоти роботи програми:***

******

******

******

**Висновок**

Отже, під час виконання лабораторної роботи №5, я ознайомився з поняттям та призначенням шаблонних функцій та шаблонних класів. Навчився використовувати шаблони класів та шаблони функцій в мові С++.