Міністерство освіти України

Національний технічний університет "ХПІ"

кафедра "Стратегічного управління і системного аналізу”

**Звіт**

**Лабораторна робота 5**

з дисципліни "Основи Програмування"

Виконав: студент групи КН-1223А

.Джим І.Р

Перевірив:

Івашко А.В.

Харків 2023

Зміст

[**Завдання** 3](#_Toc151102080)

[**1. Мета роботи** 3](#_Toc151102081)

[**2. Завдання для виконання** 3](#_Toc151102082)

[**Рішення** 3](#_Toc151102083)

[**Завдання 1** 3](#_Toc151102084)

[**Завдання 2** 4](#_Toc151102085)

[**Завдання 3** 4](#_Toc151102086)

[**Код проекту** 5](#_Toc151102087)

[**Висновок** 10](#_Toc151102088)

# **Завдання**

### **1. Мета роботи**

Метою лабораторної роботи є отримання практичних навичок в роботі з умовним оператором і розгалуженими алгоритмами в мові ***C***.

### **2. Завдання для виконання**

Треба оголосити масив цілих чисел і заповнити його випадковими значеннями. Розмір масиву і діапазон значень його елементів задані у виданому варіанті індивідуального завдання (див. п.6). В індивідуальних завданнях вказано також, яку обробку масиву слід провести.

Для всіх варіантів завдання слід мати на увазі наступне:

* нуль вважається додатним числом, якщо в завданні не обумовлений якийсь інший його статус;
* коли мова йде про якісь послідовності чисел, мається на увазі послідовність з довжиною, більшою одиниці;
* у тих випадках, коли завдання вимагає виконання якихось обчислень, дозволяється виконувати їх з тією точністю, яку забезпечують операції цілочисельної арифметики.

# **Рішення**

# **Завдання 1**

Цей код виконує декілька завдань:

1. Функція countOppositePairs:

- Приймає масив цілих чисел (`arr`) та його розмір (`size`).

- Ініціалізує змінну `count` як 0 для підрахунку пар сусідніх елементів з протилежними знаками.

- Перебирає масив з індексу 0 до `size - 1`.

- Для кожної пари сусідніх елементів перевіряє, чи одне число менше 0, а інше більше 0 (або навпаки). Якщо умова виконується, збільшує `count`.

- Повертає кількість знайдених пар.

2. Функція Task\_1:

- Ініціалізує змінну `h` для роботи з кольорами в консолі.

- Визначає розмір масиву `array` як 6.

3

- Запитує користувача ввести 6 цілих чисел для масиву.

- Виводить введені значення масиву в консоль, використовуючи різні кольори для кращої видимості.

- Викликає функцію `countOppositePairs` для підрахунку кількості пар сусідніх елементів з протилежними знаками.

- Виводить результат у консоль.

- Використовує команди `system("pause")` та `system("cls")` для затримки виконання та очищення екрану консолі після завершення виконання.

# **Завдання 2**

1. Ініціалізація та виведення місць:

- Користувацький інтерфейс виводить інформацію про доступні місця (10 місць у загальному).

- Місця для курців позначаються червоним кольором (SetConsoleTextAttribute(h, 4)), а для некурців - зеленим (SetConsoleTextAttribute(h, 2)).

2. Отримання відповіді від користувача:

- Користувачу ставиться запитання, чи він курить.

- Відповідь зчитується за допомогою `cin`.

3. Розподіл місця відповідно до відповіді:

- Якщо користувач вказав, що курить, програма спробує розподілити йому місце для курців.

- Якщо місця для курців закінчились, виводиться повідомлення про це.

- Якщо є доступні місця для курців, виводиться номер місця та відповідне повідомлення.

- Аналогічно для некурців.

4. Обробка помилкового вводу:

- Якщо користувач вводить щось інше, ніж "так" або "ні", виводиться повідомлення про помилку.

5. Завершення програми:

- Після розподілу всіх місць програма виводить повідомлення про те, що всі місця розподілені, та очищує консоль перед завершенням.

# **Завдання 3**

* 1. `HANDLE hGetStdHandle(STD\_OUTPUT\_HANDLE);`:Отримання

4

* 1. дескриптора стандартного виводу консолі для зміни кольору тексту.

2. `SetConsoleTextAttribute(h, 15);`: Встановлення кольору тексту в білий (15 - стандартний колір).

3. `int arr[12];`: Створення масиву розміром 12.

4. `for (int i = 0; i < sizeof(arr) / sizeof(int); arr[i++] = random(-100, 100));`: Заповнення масиву випадковими значеннями в межах від -100 до 100.

5. `cout << "Масив: ";`: Виведення повідомлення про початок виводу масиву.

6. `for (int i = 0; i < sizeof(arr) / sizeof(int); cout << arr[i++] << " ");`: Виведення елементів масиву на екран.

7. `int insertPosition = 0;`: Оголошення змінної для зберігання позиції вставки.

8. `int insertCount = 0;`: Оголошення змінної для зберігання кількості елементів для вставки.

9. Введення користувача для `insertPosition` та `insertCount`.

10. Перевірка коректності введених значень.

11. Введення елементів для вставки у визначену позицію.

12. `SetConsoleTextAttribute(h, 2);`: Встановлення кольору тексту в зелений (2 - зелений).

13. Виведення зміненого масиву, де вставлені елементи мають зелений колір.

14. Повернення кольору тексту в білий.

15. `system("pause");`: Зупинка виконання програми до натискання користувачем будь-якої клавіші.

16. `system("cls");`: Очищення екрану консолі.

# **Код проекту**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <iostream>

#include <stdio.h>

#include <time.h>

#include <stdlib.h>

5

#include <Windows.h>

#include <random>

#include <string>

#include <iomanip>

using namespace std;

int random(int from, int to);

enum class MenuOptions { Exit = 0, Filling, SeatsDistributor, Task3 };

void PrintMenuAndAskChoice(const HANDLE& h, MenuOptions& choice);

int countOppositePairs(int arr[], int size) {

int count = 0;

for (int i = 0; i < size - 1; ++i) {

if ((arr[i] < 0 && arr[i + 1] > 0) || (arr[i] > 0 && arr[i + 1] < 0)) {

count++;

}

}

return count;

}

void Task\_1() {

HANDLE h = GetStdHandle(STD\_OUTPUT\_HANDLE);

const int size = 6;

int array[size];

cout << "Введіть " << size << " цілих чисел:" << endl;

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

cout << "Елемент " << i + 1 << ": ";

cin >> array[i];

}

cout << "Ви ввели наступні значення у масив:" << endl;

SetConsoleTextAttribute(h, 2);

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

cout << setw(4) << array[i];

cout << endl;

SetConsoleTextAttribute(h, 15);

int result = countOppositePairs(array, size);

cout << "Кількість пар сусідніх елементів з протилежними знаками: ";

SetConsoleTextAttribute(h, 9);

cout << result << std::endl;

SetConsoleTextAttribute(h, 15);

system("pause");

system("cls");

}

void SeatsDistributor() {

HANDLE h = GetStdHandle(STD\_OUTPUT\_HANDLE);

SetConsoleTextAttribute(h, 15);

int smokingSeatsIndex = 0;

int nonSmokingSeatsIndex = 5;

int seats[10] = { 0, };

while (smokingSeatsIndex + nonSmokingSeatsIndex < 15) {

SetConsoleTextAttribute(h, 15);

cout << "Місця: ";

for (int i = 0; i < 10; i++) {

if (seats[i] == 1) {

SetConsoleTextAttribute(h, 2);

}

else {

SetConsoleTextAttribute(h, 4);

}

cout << seats[i] << " ";

}

cout << endl;

char answer[4];

SetConsoleTextAttribute(h, 15);

cout << "Палите?\n";

SetConsoleTextAttribute(h, 4);

cout << "так";

SetConsoleTextAttribute(h, 15);

cout << ",";

SetConsoleTextAttribute(h, 2);

cout << "ні";

SetConsoleTextAttribute(h, 15);

cout << "\n";

cin >> answer;

if (strcmp(answer, "так") == 0) {

if (smokingSeatsIndex >= 5) {

cout << "На жаль місця для курців закінчились\n";

continue;

}

else {

seats[smokingSeatsIndex] = 1;

smokingSeatsIndex++;

SetConsoleTextAttribute(h, 15);

cout << "Ваше місце номер ";

SetConsoleTextAttribute(h, 2);

cout << smokingSeatsIndex << endl;

}

system("cls");

}

else if (strcmp(answer, "ні") == 0) {

if (nonSmokingSeatsIndex >= 10) {

cout << "На жаль місця для не некурців закінчились\n";

continue;

}

else {

seats[nonSmokingSeatsIndex] = 1;

nonSmokingSeatsIndex++;

SetConsoleTextAttribute(h, 15);

cout << "Ваше місце номер ";

SetConsoleTextAttribute(h, 2);

cout << nonSmokingSeatsIndex << endl;

}

system("cls");

}

else {

SetConsoleTextAttribute(h, 4);

cout << "Неправильний варіант\n";

}

}

system("pause");

system("cls");

}

void ArrayInsert() {

HANDLE h = GetStdHandle(STD\_OUTPUT\_HANDLE);

SetConsoleTextAttribute(h, 15);

int arr[12];

for (int i = 0; i < sizeof(arr) / sizeof(int); arr[i++] = random(-100, 100));

cout << "Масив: ";

for (int i = 0; i < sizeof(arr) / sizeof(int); cout << arr[i++] << " ");

cout << endl;

int insertPosition = 0;

int insertCount = 0;

cout << "Введіть позицію для вставки: ";

cin >> insertPosition;

cout << "Введіть кількість елементів для вставки: ";

cin >> insertCount;

if (insertPosition > sizeof(arr) / sizeof(int) - 1 || insertPosition < 0) {

SetConsoleTextAttribute(h, 4);

cout << "Неправильне значення!\n";

SetConsoleTextAttribute(h, 15);

system("pause");

system("cls");

return;

}

if (insertPosition + insertCount > sizeof(arr) / sizeof(int) - 1) {

SetConsoleTextAttribute(h, 4);

cout << "Неправильне значення!\n";

SetConsoleTextAttribute(h, 15);

system("pause");

system("cls");

return;

}

cout << "Введіть елементи для вставки: ";

for (int i = insertPosition; i - insertPosition < insertCount; i++) {

int insert = 0;

cin >> insert;

arr[i] = insert;

}

SetConsoleTextAttribute(h, 2);

cout << "Результат: ";

SetConsoleTextAttribute(h, 15);

for (int i = 0; i < sizeof(arr) / sizeof(int); i++) {

if (i >= insertPosition && i < insertPosition + insertCount) {

SetConsoleTextAttribute(h, 2);

}

cout << arr[i] << " ";

SetConsoleTextAttribute(h, 15);

}

cout << endl;

system("pause");

system("cls");

}

void PrintMenuAndAskChoice(const HANDLE& h, MenuOptions& choice)

{

SetConsoleTextAttribute(h, 15);

printf("Виберіть програму:");

SetConsoleTextAttribute(h, 10);

//printf("\n%i", MenuOptions::Filling);

SetConsoleTextAttribute(h, 15);

printf("\nПари сусідніх значень - 1");

SetConsoleTextAttribute(h, 10);

//printf("\n%i", MenuOptions::SeatsDistributor);

SetConsoleTextAttribute(h, 15);

printf("\nМісця в літаку - 2");

SetConsoleTextAttribute(h, 10);

//printf("\n%i", MenuOptions::Task3);

SetConsoleTextAttribute(h, 15);

printf("\nВставка в масив - 3");

SetConsoleTextAttribute(h, 10);

//printf("\n%i", MenuOptions::Exit);

SetConsoleTextAttribute(h, 15);

printf("\nВихід - 0 \nПрограма: ");

SetConsoleTextAttribute(h, 6);

scanf("%i", &choice);

SetConsoleTextAttribute(h, 15);

}

int main()

{

setlocale(LC\_ALL, "Russian");

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

HANDLE h = GetStdHandle(STD\_OUTPUT\_HANDLE);

SetConsoleTextAttribute(h, 15);

MenuOptions choice;

while (true) {

PrintMenuAndAskChoice(h, choice);

switch (choice)

{

case MenuOptions::Filling:

Task\_1();

break;

case MenuOptions::SeatsDistributor:

SeatsDistributor();

break;

case MenuOptions::Task3:

ArrayInsert();

break;

case MenuOptions::Exit:

SetConsoleTextAttribute(h, 4);

cout << "Завершення...\n";

SetConsoleTextAttribute(h, 15);

system("pause");

system("cls");

return 0;

default:

SetConsoleTextAttribute(h, 4);

printf("Неправильне значення!\n");

SetConsoleTextAttribute(h, 15);

system("pause");

system("cls");

break;

}

}

}

int random(int from, int to) {

if (from > 0) {

int result = rand() % (from - to + 1) + to;

return result;

}

if (from < 0) {

int result = rand() % (to - from + 1) + from;

return result;

}

}

# **Висновок**

В ході виконання лабораторної роботи у середовищі Visual Studio були отримані важливі навички програмування та роботи з інтегрованим середовищем розробки. За допомогою мови програмування, яка була використана в цій лабораторній роботі, вдалося створити програму, яка відповідає поставленим завданням.

Одним з ключових вмінь, які були розвинуті, є вміння розробляти і відлагоджувати код. Під час лабораторної роботи було необхідно виявити та виправити помилки в програмі, що сприяє покращенню розуміння процесу відлагодження.

Також, лабораторна робота дозволила ознайомитися з основними концепціями створення графічного інтерфейсу користувача, що є важливим аспектом розробки програм. Використання графічного інтерфейсу полегшило взаємодію з користувачем та зробило програму більш зручною та доступною.

Під час виконання лабораторної роботи було надано можливість вивчити та використовувати різноманітні бібліотеки та інструменти, які розширюють можливості Visual Studio. Це дозволяє покращити продуктивність та ефективність розробки.

Завдяки цій лабораторній роботі, ми також набули досвіду у спільній роботі над програмним проектом, що може бути корисним у майбутньому професійному житті.

У підсумку, виконання цієї лабораторної роботи дало нам можливість розширити свої знання та вміння в області програмування, відлагодження коду, створення графічного інтерфейсу та спільної роботи над проектами. Все це є важливими компонентами розвитку як професійного програміста.

10