

Міністерство освіти і науки України

Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії

*Кафедра програмної інженерії*

## ЗВІТ

до лабораторної роботи №3

з навчальної дисципліни «Алгоритми і структури даних»

Тема: «Абстрактні типи даних «стек» та «черга»»

Перевірив:

Стоянов Ю. М.

Підготував:

студент групи СП-12

Штокало А. Р.

Тернопіль 2023

Мета: набути практичних навичок по реалізації та роботі зі АДТ «стек» та «черга».

Завдання:

1. Реалізувати на мові C/C++ представлення АДТ «стек» при допомозі масиву (операції MAKENULL, PUSH, POP, TOP, EMPTY, PRINT).
2. Реалізувати на мові C/C++ представлення АДТ «черга» при допомозі циклічного масиву (операції MAKENULL, FRONT, ENQUEUE, DEQUEUE, EMPTY).

IDE: Microsoft Visual Studio 2022

Завдання 1

Лістинг програми:

```
#include <iostream>
#include <Windows.h>

using namespace std;

struct Stack {
    int* arr;
    int top = -1;
    int capacity;
};

void MAKENULL(Stack& s) {
    s.top = -1;
}

void PUSH(int x, Stack& s) {
    s.top++;
    s.arr[s.top] = x;
}

void POP(Stack& s) {
    if (s.top == -1) {
        cout << "Стек порожній!" << endl;
```

```

        return;
    }
    s.top--;
}

int TOP(Stack& s) {
    if (s.top == -1) {
        cout << "Стек порожній!" << endl;
        return -1;
    }
    return s.arr[s.top];
}

bool EMPTY(Stack& s) {
    return s.top == -1;
}

void PRINT(Stack& s) {
    if (s.top == -1) {
        cout << "Стек порожній!" << endl;
        return;
    }
    cout << "Стек: ";
    for (int i = s.top; i >= 0; i--) {
        cout << s.arr[i] << " ";
    }
    cout << endl;
}

int main() {
    SetConsoleCP(1251);
    SetConsoleOutputCP(1251);
    Stack s;
    cout << "Введіть розмірність стеку: ";
    cin >> s.capacity;

    s.arr = new int[s.capacity];

```

```

cout << "Введіть елементи стеку: ";
for (int i = 0; i < s.capacity; i++) {
    int x;
    cin >> x;
    PUSH(x, s);
}

bool exit = false;
while (!exit) {
    cout << "Оберіть операцію:" << endl;
    cout << "1. MAKENULL" << endl;
    cout << "2. PUSH" << endl;
    cout << "3. POP" << endl;
    cout << "4. TOP" << endl;
    cout << "5. EMPTY" << endl;
    cout << "6. PRINT" << endl;
    cout << "7. Вихід" << endl;

    int choice;
    cin >> choice;

    switch (choice) {
    case 1:
        MAKENULL(s);
        PRINT(s);
        break;
    case 2:
        int x;
        cout << "Введіть елемент: ";
        cin >> x;
        PUSH(x, s);
        PRINT(s);
        break;
    case 3:
        POP(s);
        PRINT(s);

```

```

        break;
    case 4:
        cout << "Вершина стеку: " << TOP(s) << endl;
        break;
    case 5:
        cout << "Стек порожній? " << EMPTY(s) << endl;
        break;
    case 6:
        PRINT(s);
        break;
    case 7:
        exit = true;
        break;
    default:
        cout << "Невірний вибір. Спробуйте ще раз!" << endl;
        break;
    }
}

delete[] s.arr;

return 0;
}

```

## Результат:

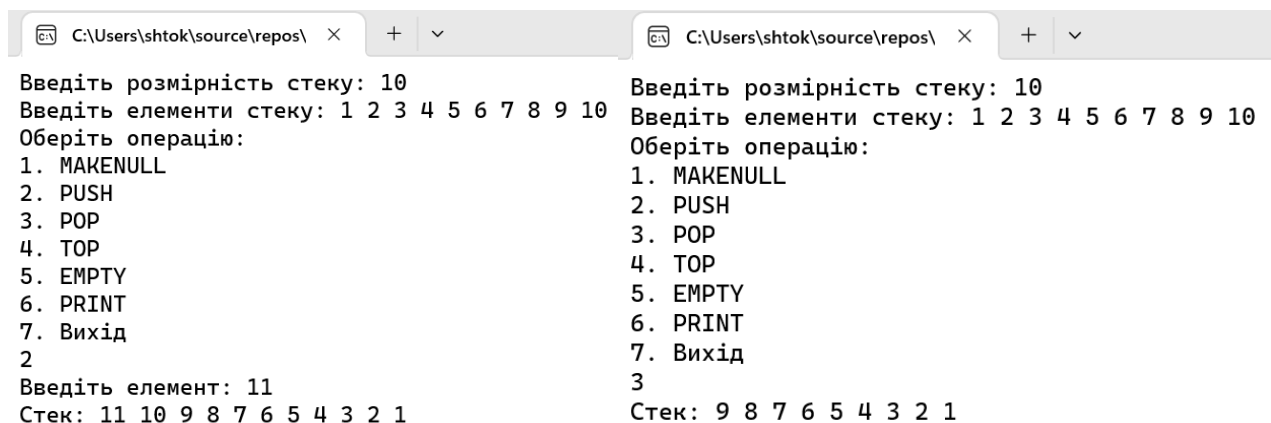


Рисунок 1 - Результат тестування завдання 1

## Завдання 2

### Лістинг програми:

```
#include <iostream>
```

```

#include <Windows.h>

using namespace std;

const int MAX_SIZE = 100;

struct Queue {
    int data[MAX_SIZE];
    int front;
    int rear;
    int size;
};

void MAKENULL(Queue& Q) {
    Q.front = 0;
    Q.rear = MAX_SIZE - 1;
    Q.size = 0;
}

int FRONT(Queue& Q) {
    return Q.data[Q.front];
}

void ENQUEUE(int x, Queue& Q) {
    if (Q.size == MAX_SIZE) {
        cout << "Переповнення черги" << endl;
        return;
    }
    Q.rear = (Q.rear + 1) % MAX_SIZE;
    Q.data[Q.rear] = x;
    Q.size++;
}

void DEQUEUE(Queue& Q) {
    if (Q.size == 0) {
        cout << "Недостатньо елементів у черзі" << endl;
        return;
    }

```

```

    }

    Q.front = (Q.front + 1) % MAX_SIZE;
    Q.size--;
}

bool EMPTY(Queue& Q) {
    return Q.size == 0;
}

void PRINT(Queue& Q) {
    if (Q.size == 0) {
        cout << "Черга порожня" << endl;
        return;
    }
    cout << "Черга: ";
    for (int i = Q.front; i <= Q.rear; i = (i + 1) % MAX_SIZE) {
        cout << Q.data[i] << " ";
    }
    cout << endl;
}

int main() {
    SetConsoleCP(1251);
    SetConsoleOutputCP(1251);

    Queue Q;
    int size;
    cout << "Введіть розмір черги: ";
    cin >> size;
    MAKENULL(Q);
    cout << "Введіть елементи черги: ";
    for (int i = 0; i < size; i++) {
        int x;
        cin >> x;
        ENQUEUE(x, Q);
    }
    while (true) {

```

```
int choice;
cout << endl << "Оберіть операцію:" << endl;
cout << "1. MAKENULL" << endl;
cout << "2. FRONT" << endl;
cout << "3. ENQUEUE" << endl;
cout << "4. DEQUEUE" << endl;
cout << "5. EMPTY" << endl;
cout << "6. PRINT" << endl;
cout << "0. Вихід" << endl;
cin >> choice;
switch (choice) {
case 0:
    return 0;
case 1:
    MAKENULL(Q);
    break;
case 2:
    if (EMPTY(Q)) {
        cout << "Черга порожня" << endl;
    }
    else {
        cout << "Перший елемент: " << FRONT(Q) << endl;
    }
    break;
case 3:
    int x;
    cout << "Введіть елемент для включення: ";
    cin >> x;
    ENQUEUE(x, Q);
    break;
case 4:
    DEQUEUE(Q);
    break;
case 5:
    if (EMPTY(Q)) {
        cout << "Черга порожня" << endl;
    }
}
```



```

        else {
            cout << "Черга не порожня" << endl;
        }
        break;
    case 6:
        PRINT(Q);
        break;
    default:
        cout << "Невірний вибір. Спробуйте ще раз!" << endl;
    }
}
return 0;
}

```

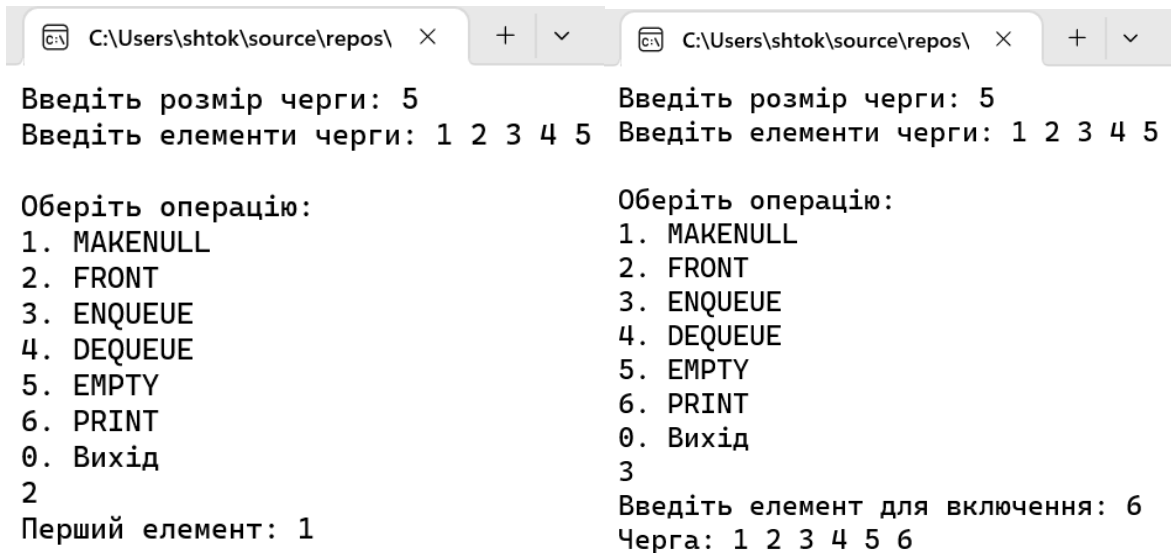


Рисунок 2 - Результат тестування завдання 2

## Висновок:

Під час виконання роботи з метою набуття практичних навичок по реалізації та роботі зі стеком та чергою, я можу зробити висновок, що було успішно досягнуто мету. В процесі роботи я навчився реалізовувати структури даних "стек" та "черга" та застосовувати їх у різних задачах. Я також отримав практичний досвід в роботі з цими АТД, що дозволить мені застосовувати їх у майбутніх проектах. Загалом, ця мета була досягнута, і я здобув корисні знання та навички у галузі розробки програмного забезпечення.