

# Лабораторна робота №11. Шаблонні класи

Тема: Шаблонні класи.

Мета: Отримати базові знання про шаблонізацію (узагальнення) на основі шаблонних класів.

## ВИМОГИ

### 1.1 Інформація про розробника:

- Кліщов Б. Р.
- КІТ 102.8а

### 1.2 Загальне завдання

Модернізувати клас, що був розроблений у попередній роботі наступним шляхом:

- зробити його шаблонним;
- додати поле – шаблонний масив;
- видалити з аргументів існуючих методів масив, а замість цього використовувати масив-поле класу. Необхідно продемонструвати роботу програми як з використанням стандартних типів даних, так і типів, які створені користувачем.

### 1.3 Додаткові умови виконання завдання:

- продемонструвати відсутність витоків пам'яті;
- продемонструвати роботу розроблених методів за допомогою модульних тестів;
- не використовувати конструкцію «`using namespace std;`», замість цього слід роботи «`using`» кожного необхідного класу: `using std::string;` `using std::cout.`

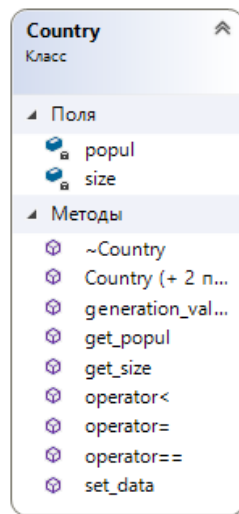
## 2. ОПИС ПРОГРАМИ

### 2.1 Функціональне призначення

Програма призначена щоб отримувати та зберігати інформацію щодо різних країн світу.

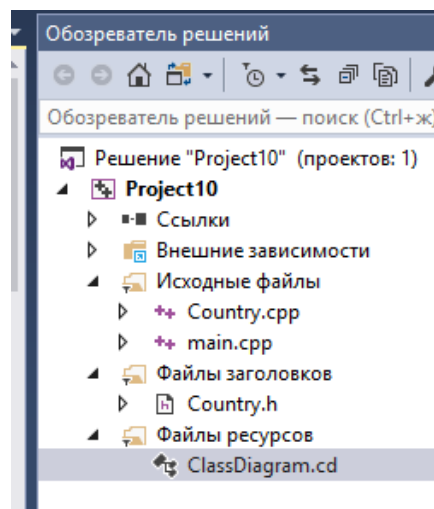
## 2.2 Опис логічної структури

На рисунку № 1 зображена діаграма класу



Малюнок №1. Діаграма класу

На рисунку № 2 зображена структура програми:



Малюнок №2: Структура програми

## 2.3 Важливі фрагменти програми

Код програми :

```
template <class T>
class Functions {
private:
    T* arr;
    size_t size;
public:
    Functions() : size(0) {}

    Functions(size_t size) : size(size) {
        arr = new T[size];
    }

    void enterData() {
        std::cout << "Enter marks: " << std::endl;
        for (int i = 0; i < size; i++) {
            std::cin >> arr[i];
        }
    }

    void printArr() {
        for (int i = 0; i < size; i++) {
            std::cout << arr[i] << std::endl;
        }
    }

    T minimum() {
        T min = arr[0];
        for (int i = 0; i < size; i++) {
            if (arr[i] < min) {
                min = arr[i];
            }
        }
        return min;
    }

    void sortArr() {
        T temp;
        for (int i = 0; i < size; i++) {
            for (int j = 0; j < size; j++) {
                if (arr[i] < arr[j]) {
                    temp = arr[i];
                    arr[i] = arr[j];
                    arr[j] = temp;
                }
            }
        }
    }

    int getByIndex(T values) {
        for (int i = 0; i < size; i++) {
            if (arr[i] == values) {
                return i;
            }
        }
        return -1;
    }

    ~Functions() {
        delete[] arr;
    }
}
```

```

};

int main() {

    int size = 0;
    std::cout << "Enter size : ";
    std::cin >> size;
    int* arr = new int[size];

    {
        int values;
        Functions<int> work;
        work.enterData();
        int option;
        do {
            std::cout << "Choose option:" << std::endl << "0 - Exit " <<
std::endl << "1 - Print array" << std::endl << "2 - Sort array" << std::endl << "3 -
Search by index" << std::endl << "4 - Search by min values" << std::endl;
            std::cout << std::endl;
            std::cin >> option;

            switch (option) {
            case 1: {
                system("cls");
                work.printArr();
                break;
            }
            case 2: {
                system("cls");
                work.sortArr();
                work.printArr();
                break;
            }
            case 3: {
                std::cout << "Enter values: ";
                std::cin >> values;
                system("cls");
                work.getByIndex(values);
                std::cout << values << std::endl << std::endl;
                break;
            }
            case 4: {
                int min = work.minimum();
                system("cls");
                std::cout << min << std::endl << std::endl;
                break;
            }
            default: {
                break;
            }
        }
        } while (option != 0);
        system("cls");
    }
    delete[]arr;

    _CrtSetReportMode(_CRT_WARN, _CRTDBG_MODE_FILE);
    _CrtSetReportFile(_CRT_WARN, _CRTDBG_FILE_STDERR);
    _CrtSetReportMode(_CRT_ERROR, _CRTDBG_MODE_FILE);
    _CrtSetReportFile(_CRT_ERROR, _CRTDBG_FILE_STDERR);
    _CrtSetReportMode(_CRT_ASSERT, _CRTDBG_MODE_FILE);
    _CrtSetReportFile(_CRT_ASSERT, _CRTDBG_FILE_STDERR);
    _CrtDumpMemoryLeaks();
    system("pause");
    return _CrtDumpMemoryLeaks();
}

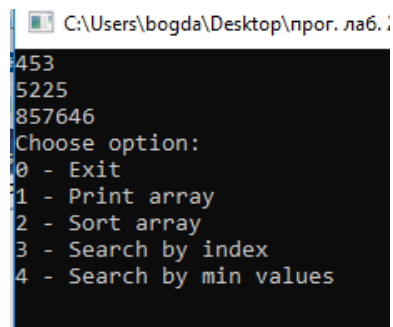
```

}

### 3 ВАРІАНТИ ВИКОРИСТАННЯ

#### 3.1 Результат роботи функцій

На рисунку № 3 зображено результат роботи програми



```
C:\Users\bogda\Desktop\npor. лаб. 1
453
5225
857646
Choose option:
0 - Exit
1 - Print array
2 - Sort array
3 - Search by index
4 - Search by min values
```

Рисунок № 3. Результат роботи програми

**Висновок:** Отримати базові знання про шаблонізацію (узагальнення) на основі шаблонних класів.