

## Лабораторна робота №12. STL

Тема. STL. Ітератори. Послідовні контейнери. Цикл range-for. Асоціативні контейнери.

Мета. Отримати базові знання про STL контейнери. Освоїти основні механізми роботи з STL контейнерами.

### 1 Вимоги

#### 1.1 Розробник:

1. Кононенко Дмитро Олексійович
2. НТУ “ХПІ”,
3. КІТ 102.8а

#### 1.2 Завдання

Загальне завдання. Маючи класи з прикладної області РГЗ (тільки базовий клас та клас/класи спадкоємці), створити діалогове меню, що дозволяє продемонструвати роботу STL контейнерів (додавання / видалення / отримання даних, показ всіх елементів) та показати їх принципову різницю:

- vector;
- set;
- list;
- map .

При цьому врахувати, що контейнери містять елементи одного типу, наприклад, базового. Прохід по всьому контейнеру повинен виконуватися за допомогою циклу мови C++11 – range-for.

Додаткове завдання на оцінку «відмінно»:

контейнери повинні оперувати даними не тільки базового класу, а ще даними класів-спадкоємців.

## 2. Опис програми

### 2.1 Призначення:

Програма розроблена для ознайомлення з STL бібліотекою.

#### 2.1.1 Опис функцій для класу Map:

1. Функція додавання
2. Функція виводу на екран
3. Функція пошуку
4. Функція видалення

#### 2.1.2 Опис функцій для класу List:

1. Функція додавання
2. Функція виводу на екран
3. Функція пошуку
4. Функція видалення

#### 2.1.3 Опис функцій для класу Set:

1. Функція додавання
2. Функція виводу на екран
3. Функція пошуку
4. Функція видалення

#### 2.1.4 Опис функцій для класу Vector:

1. Функція додавання
2. Функція виводу на екран
3. Функція пошуку
4. Функція видалення

## 2.2 Приклади функцій:

```
int i = 0;
int j, choose;
set<T> qual;
T tmp;
int size;

do {
    cout << "\n";
    cout << "0 - Exit" << endl;
    cout << "1 - Print vector" << endl;
    cout << "2 - Find element" << endl;
    cout << "3 - Add element" << endl;
    cout << "4 - Delete element" << endl;
    cout << "Choose: ";
    cin >> choose;

    switch (choose)
    {
    case 1:
        for (auto &it : qual) { //цикл for_each
            cout << it;
        }
        break;
    case 2:
        cout << "Input search element: " << endl;
        cin >> tmp;

        if (qual.find(tmp) != qual.end()) {
            cout << "Exist: " << tmp << endl;
        }
    }
```

Рисунок 1 — функції пошуку та виводу

```
313 case 3: {
314     cout << "Input index: ";
315     cin >> key;
316     cout << "Input new element: " << endl;
317     cin >> tmp;
318     qual.emplace(key,tmp);
319     break;
320 }
321 case 4:
322     cout << "input delete point: ";
323     cin >> key;
324
325     if (key < 0 || key > qual.size()) {
326         break;
327     }
328
329     qual.erase(key);
330     break;
331 default:
```

Рисунок 2 — функції видалення та додавання

Висновок: навчився працювати з бібліотекою STL. Визначив різницю між такими шаблонними класами як: Vector, Map, Set, List. Vector має найшвидший доступ до об'єктів, Map зберігає ключ та інформацію(будь-який тип даних, ключі не можуть повторюватися), Set зберігає будь-яку інформацію(але об'єкти не будуть повторюватися, а також вони будуть відсортовані), List - безмежний розмір.