

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

- Котенко Сергій Миколайович;
- Студент групи КІТ 102.8(а);
- 09-06-2019р..

1.2 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»:

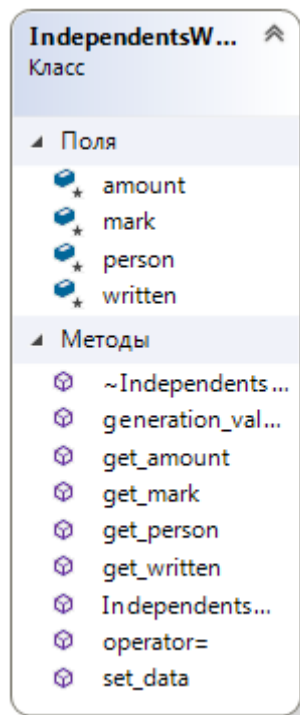
Контейнери повинні оперувати даними не тільки базового класу, а ще даними класів-спадкоємців.

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 Опис логічної структури

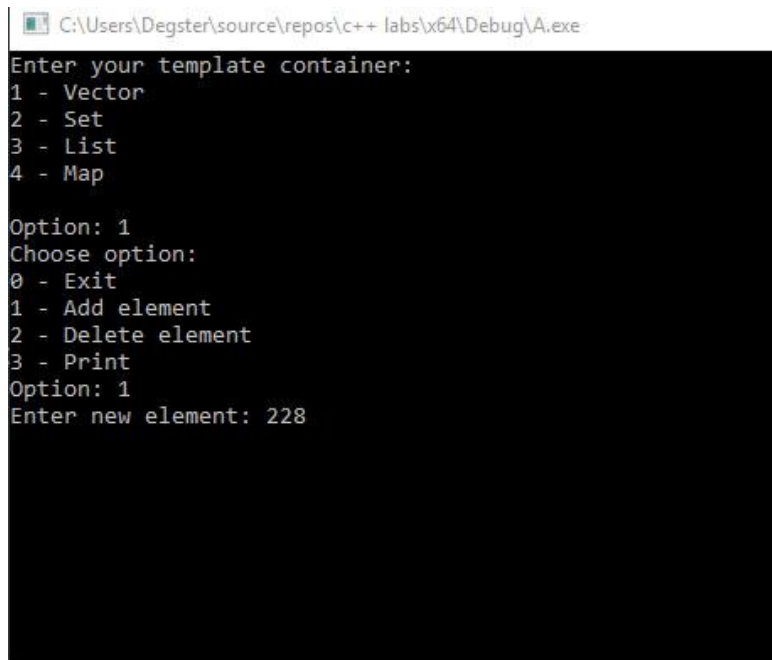


Діаграма класу `IndependentsWork`:

- ✓ `~IndependentsWork` - Деструктор класу;
- ✓ `generation_values` – Генерація випадкових значень;
- ✓ `get_amount`, `get_mark`, `get_person`, `get_written` - Отримання даних;
- ✓ `InfoIndependentsWork` - Конструктор класу;
- ✓ `operator=` - Перевантаження оператора;
- ✓ `set_data` - Встановлення значень .

3 ВАРІАНТИ ВИКОРИСТАННЯ

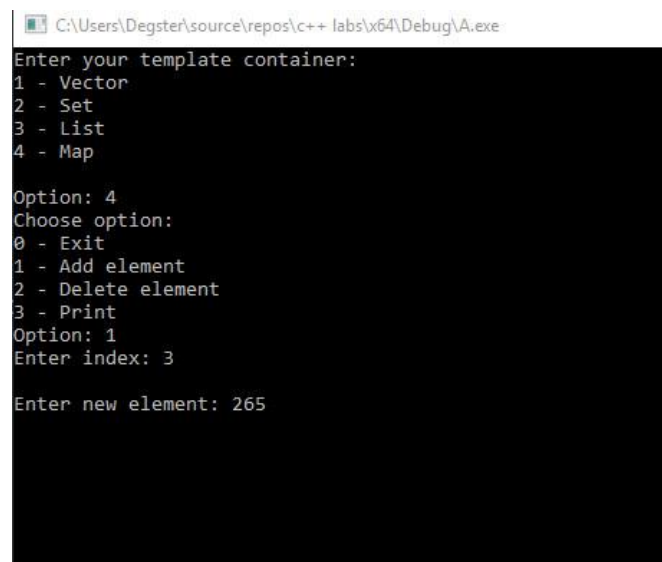
3.1 Ілюстрація роботи програми



```
C:\Users\Degster\source\repos\c++ labs\64\Debug\A.exe
Enter your template container:
1 - Vector
2 - Set
3 - List
4 - Map

Option: 1
Choose option:
0 - Exit
1 - Add element
2 - Delete element
3 - Print
Option: 1
Enter new element: 228
```

Рисунок 3.1 – Вибір роботи з контейнером та можливі функції



```
C:\Users\Degster\source\repos\c++ labs\64\Debug\A.exe
Enter your template container:
1 - Vector
2 - Set
3 - List
4 - Map

Option: 4
Choose option:
0 - Exit
1 - Add element
2 - Delete element
3 - Print
Option: 1
Enter index: 3
Enter new element: 265
```

Рисунок 3.2 – Додавання нового елементу

ВИСНОВОК

В інтегрованому середовищі *Visual Studio* розроблена програма мовою C++. Виконання програми дозволяє продемонструвати коректність роботи контейнерів.