ЛАБОРАТОРНА РОБОТА №13.

АЛГОРИТМИ ПЕРЕМІЩЕННЯ ТА ПОШУКУ

*Тема.* STL. Алгоритми переміщення та пошуку.

*Мета.* На практиці порівняти STL алгоритми, що не модифікують послідовність.

1 ВИМОГИ

**1.1 Розробник**

- Котенко Сергій Миколайович;

- Студент групи КІТ 102.8(а);

- 09-06-2019р..

**1.2 Загальне завдання**

Поширити попередню лабораторну роботу, додаючи наступні можливості діалогового меню:

- вивід всіх елементів масиву за допомогою STL функції for\_each;

- визначення кількості елементів за заданим критерієм;

- пошук елементу за заданим критерієм.

Додаткові умови виконання завдання:

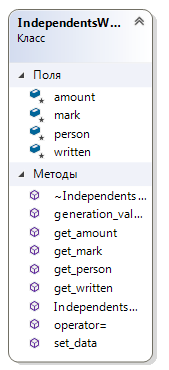
- Продемонструвати відсутність витоків пам’яті

- Продемонструвати роботу розроблений 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 операторa;
* set\_data - Встановлення значень .

**2.2 Фрагменти коду**

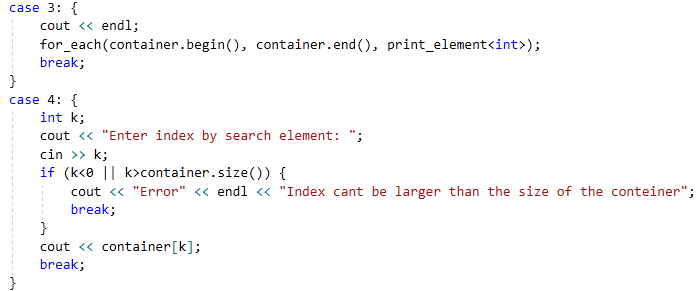


Рисунок 2.1 – Функція for\_each та пошук за заданим критерієм

**3 ВАРІАНТИ ВИКОРИСТАННЯ**

**3.1 Ілюстрація роботи програми**

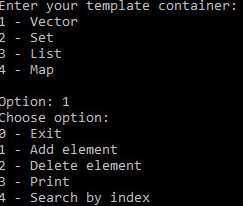


Рисунок 3.1 – Вибір роботи з контейнером та можливі функції

**ВИСНОВОК**

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