

Лабораторна робота № 13

Тема. STL. Алгоритми переміщення та пошуку.

Мета роботи. На практиці порівняти алгоритми, що не модифікують послідовність.

1 ВИМОГИ

1.1 Розробник

Інформація про розробника:

- Кулик Данііл Ігорович
- НТУ “ХПІ” КІТ 102.8а
- Варіант 12

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 елемента за заданим критерієм.

2 ОПИС ПРОГРАМИ

2.1 Функціональне призначення

За допомогою цієї програми можна створити масив об'єктів, додавати та видаляти об'єкти, виводити вміст масиву на екран та вивід об'єкта по індексу. Також у цій програмі реалізоване зручне меню спілкування з користувачем.

2.2 Важливі фрагменти програми

На рисунку №1 зображено діаграми класів.

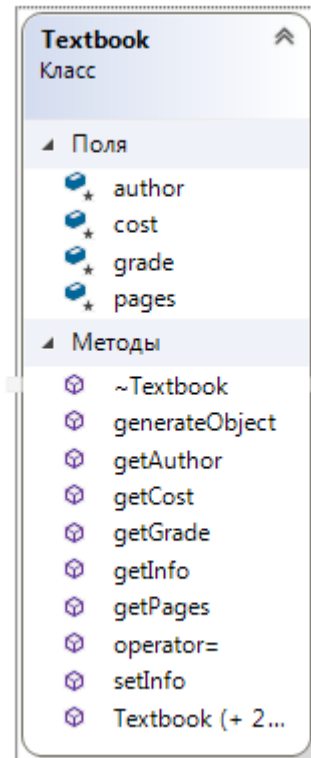


Рисунок №1 – діаграма класу Textbook

Методи класу **Textbook**:

- **Textbook()** - Конструктор за замовчуванням ;
- **Textbook(int pages, int grade, int cost, string author)** – конструктор з параметрами;
- **Textbook(const Textbook &obj)** – конструктор копіювання ;
- **void setInfo(int pages, int grade, int cost, string author)** – заповнення поля **Textbook:: pages**(інші методи **get** роблять теж саме, але з іншими полями);
- **int getPages()const** – читання значення поля **pages** (інші методи **get** роблять теж саме, але з іншими полями);
- **Textbook& operator = (Textbook &obj)** - перевантажений оператор присвоювання.

На рисунках №2 та №3 зображено наявність STL контейнерів.

```

68 void listMethod() {
69     int index, option;
70     list<T> qual;
71     T tmp;
72
73     do { ... } while (option != 0);
129
130     qual.clear();
131     return;
132 }
133
134 template<class T>
135 void vectorMethod() {
136     ForClass<Textbook> obj;
137     int index, choose;
138     vector<T> qual;
139     vector<T> qual2;
140     T tmp;
141     int size;
142     cout << "Enter size of vector: ";
143     cin >> size;
144
145     for (int i = 0; i < size; i++) {
146         cin >> tmp;
147         qual2.push_back(tmp);
148     }
149
150     do { ... } while (choose != 0);
196
197     qual.clear();
198     qual2.clear();
199     return;
200 }

```

Рисунок №2 – STL контейнеры *list*, *vector*.

```

202 template<class T>
203 void setMethod() {
204     int i = 0;
205     int j, choose;
206     set<T> qual;
207     T tmp;
208     int size;
209
210     do { ... } while (choose != 0);
270
271     return;
272 }
273
274
275 template<class T> <T>
276 void mapMethod() {
277     int key, choose;
278     map<int, T> qual;
279     T tmp;
280     int size;
281
282     do { ... } while (choose != 0);
332
333     return;
334 }

```

Рисунок №3 – STL контейнеры *set*, *map*.

3 ВАРІАНТИ ВИКОРИСТАННЯ

Програма може бути використана для створення масиву об'єктів. Програма має методи додавання, видалення об'єктів, вивід об'єкта по індексу з масиву та усіх об'єктів на екран, читання та запис об'єктів масиву з файлу, а також сортування за одним із властивостей об'єкта. Меню робить роботу з цією програмою зручною.

1. Меню спілкування з користувачем зображено на рисунку №4

```
Enter size: 3
Enter number of pages:
228
322
160

Welcome to our menu list. What do you want?:
0 - Exit
1 - Print array
2 - Sort array
3 - Search by index
4 - Search by min values
```

Рисунок №4 – меню спілкування з користувачем

2. Результат виводу об'єктів після сортування зображено на рисунку №5.

```
Number of pages in 1 textbook is: 160
Number of pages in 2 textbook is: 228
Number of pages in 3 textbook is: 322

Welcome to our menu list. What do you want?:
0 - Exit
1 - Print array
2 - Sort array
3 - Search by index
4 - Search by min values
```

Рисунок №5 - результат виводу об'єктів масиву після сортування

ВИСНОВКИ

В інтегрованому середовищі VisualStudio розроблена програма мовою C++. Засоби налагодження дозволяють за допомогою меню спілкування створити масив об'єктів та змінювати його за допомогою методів класів.