

ПРАКТИЧЕСКАЯ РАБОТА №2

Последовательные контейнеры STL

1. Цель работы

Изучить последовательные контейнеры и способы работы с ними.

2. Содержание работы

1. Изучить элементы стандартной библиотеки шаблонов C++: последовательные контейнеры, адаптеры контейнеров, функциональные объекты, функциональные адаптеры, лямбда-выражения.

2. Переработать программу из работы №1 так, чтобы объекты разработанного класса хранились в одном из следующих контейнеров STL: список (list), односвязный список (forward_list). Использовать функции-члены контейнера для реализации следующих возможностей:

2.1. Добавление объектов в контейнер.

2.2. Поиск объектов в контейнере по различным критериям.

2.3. Модификация объектов в контейнере (возможно, в соответствии с некоторыми критериями).

2.4. Удаление объектов из контейнера по различным критериям.

2.5. Вывод всех объектов контейнера, отсортированных по различным критериям, в файл и/или на экран.

3. При формировании критериев поиска, удаления, сортировки использовать функциональные объекты (пользовательские и/или стандартные) и/или лямбда-выражения. Атрибуты класса, хранящиеся как строки, но по смыслу являющиеся числами, должны участвовать в сравнениях как числа.

4. Оформить отчет. Отчет должен содержать постановку задачи, алгоритм, описание и текст разработанной программы, результаты тестирования (со скриншотами) и выводы.

5. Защитить работу, ответив на вопросы преподавателя.

3. Контрольные вопросы

Контейнеры

1. Возможности контейнеров.
2. Операции над контейнерами.
3. Массив (array).
4. Вектор (vector).
5. Дек (deque).
6. Список (list).
7. Односвязный список (forward_list).
8. Адаптеры контейнеров.

Функциональные объекты и лямбда-выражения

9. Концепция функциональных объектов.

10. Стандартные функциональные объекты и функциональные адаптеры.

11. Лямбда-выражения.