**STL: ПОСЛЕДОВАТЕЛЬНЫЕ КОНТЕЙНЕРЫ**

***Часть 1: теоретическая –вопросы для ответа при сдаче работы.***

По каждому вопросу из списка подготовить демонстрирующий пример.

**VECTOR**

1. Какие виды векторов позволяют создать конструкторы?

2. Как задать размер контейнера?

3. Как определить потенциальный размер контейнера?

4. Как получить доступ к элементу контейнера?

5. Как присвоить значение элементу контейнера?

6. Опишите функциональность методов begin и end

7. Опишите использование метода back и front

8. Опишите использование метода insert

9. Как и куда можно вставить новые элементы в контейнер?

10. Опишите функциональность методов capacity и max\_size

11. Зачем нужны методы reserve и resize?

12. Как вставит/удалить элемент в начало контейнера?

13. Как вставит/удалить элемент в конец контейнера.

**DEQUE**

14. Какие виды двусторонних очередей позволяют создать конструкторы?

15. Как получить доступ к элементу контейнера?

16. Как присвоить значение элементу контейнера?

17. Опишите функциональность методов front и back

18. Как вставит/удалить элемент в начало контейнера?

19. Как вставит/удалить элемент в конец контейнера.

**LIST**

20. Опишите использование метода merge.

21. Опишите использование метода remove\_if.

22. Опишите использование метода splice.

23. Опишите использование метода unique.

**ОБЩИЕ ВОПРОСЫ**

24. У каких контейнеров допустим произвольный доступ к элементам?

25. Для каких контейнеров сохраняются значения указателей, итераторов после вставки/удаления?

26. Какие последовательные контейнеры поддерживают упорядоченность элементов автоматически?

27. Какие средства можно использовать для сортировки элементов контейнера?

28. Когда, в каких случаях нужно отдать предпочтение выбору одного из контейнеров?

29. Какие средства можно использовать и что необходимо реализовать для сравнения элементов контейнеров, если они содержат объекты пользовательских классов?

***Часть 2: решение задач.***

**Задание №1**

*Просмотр матрицы с итераторами!!!!!*

Работа с матрицей vector<vector<int>>, размеры и максимум вводятся с клавиатуры.

Задание:

- заполнить случайными значениями;

- удалить строку и столбец, на пересечении которых стоит максимальный элемент (первый из найденных);

- упорядочить строки матрицы по возрастанию сумм элементов строк (без использования алгоритма sort).

**Задание №2**  

Каждая строка текстового файла “Students.txt” содержит информацию об одном студенте: ***ФИО курс группа***. Файл большой, и заранее не известно количество информации, содержащейся в нем.

Необходимо получить два новых текстовых файла.

Первый “Fio.txt” должен содержать все данные о студентах, упорядоченные по фамилиям по алфавиту.

Второй “Groups.txt” - данные, упорядоченные по курсам, для одинакового курса по группам, для одинаковой группы по алфавиту.

**Использовать контейнер vector.**

В файле “Students.txt” данные не должны быть упорядочены, они должны быть перемешаны. Файл “Students.txt” должен содержать сведения о 3 курсах, на каждом курсе по 2 группы и в каждой группе по 4 студента.

**STL: АССОЦИАТИВНЫЕ КОНТЕЙНЕРЫ**

***Часть 1: теоретическая –вопросы для тестирования.***

**По каждому вопросу из списка подготовить демонстрирующий пример.**

1. С какой целью и как используется класс pair?

2. Запишите каким образом можно создавать пары?

3. Дайте характеристику параметрам контейнера map.

4. Доступ к элементам map.

5. Какие итераторы можно использовать с map?

6. Методы поиска в map.

7. Что произойдет, если добавляется в map элемент с существующим ключом?

8. Вставка элементов в map.

9. Удаление элементов из map.

10. Использование методов lower\_bound и upper\_bound.

11. Использование метода equal\_range.

12. Дайте характеристику параметрам контейнера multimap.

13. Методы поиска в multimap.

14. Что произойдет, если добавляется map элемент с существующим ключом?

15. Вставка элементов в multimap (ключи совпадают\не совпадают).

16. Удаление элементов из multimap (ключи совпадают\не совпадают).

17. Использование методов lower\_bound и upper\_bound.

18. Использование метода equal\_range.

***Часть 2: решение задач.***

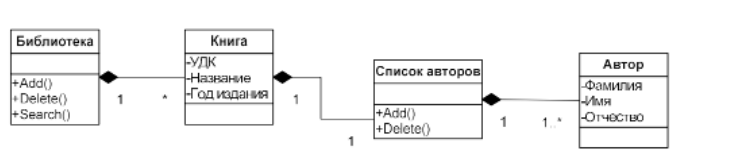
**Задание №1.** «Вложенные контейнеры: список в списке». Под списком имеется ввиду перечень. Выбирайте наиболее подходящие контейнеры.

Составить программу, которая содержит текущую информацию о книгах в библиотеке. Сведения о книге содержат:

* номер УДК (целое число);
* список авторов (хранится фамилия, имя, отчество; количество авторов не ограничено);
* название;
* год издания.

Книги в списке должны храниться упорядоченными по названию, авторы – по фамилиям.

Информация о книгах считывается из текстового файла, в котором она хранится в неупорядоченном виде.



Требуется реализовать:

* добавление книги,
* удаление книги,
* поиск книг по названию,
* поиск книг указанного автора,
* добавление и удаления автора указанной книги.