**Лабораторная работа №4**

Цель работы: Изучение принципов организации и работы с абстрактной структурой данных СТЭК в форме односвязного линейного списка и с абстрактной структурой данных ОЧЕРЕДЬ в форме односвязного линейного списка.

**Задание 1**

Разработать консольное приложение, которое с помощью абстрактной структуры данных СТЭК проверяет соответствие открывающих и   
закрывающих HTML-тэгов во фрагменте HTML кода, введённого с клавиатуры.

Приложение должно:

1. делать запрос на ввод HTML кода;

2. выводить в консоль:

- в случае соответствия – сообщение об этом;

- в случае несоответствия – сообщение об этом и тэг(и), для которого нет пары;

3. делать запрос на выход из приложения.

**Задание 2**

-Даны две непустые очереди, которые содержат одинаковое количество элементов. Объединить очереди в одну, в которой элементы исходных очередей чередуются.

-Даны две непустые очереди. Элементы каждой из очередей упорядочены по возрастанию. Объединить очереди в одну с сохранением упорядоченности элементов.

**Альтернативное задание**

Разработать консольное приложение, которое с помощью абстрактной структуры данных ОЧЕРЕДЬ моделирует работу аэропорта (или другой системы массового обслуживания) с одной взлётно-посадочной полосой, которую в каждый момент времени (цикл for) может использовать только один самолёт – для взлёта или посадки.

Приложение должно:

1. сделать запрос на ввод данных для моделирования:

- максимальное кол-во самолётов в очереди на посадку и взлёт;

- интервал времени для моделирования;

- предполагаемое (ожидаемое) число прилетающих и взлетающих самолётов в единицу времени;

2. используя псевдослучайные числа, удовлетворяющие закону

распределения Пуассона, смоделировать кол-во запросов на взлёт

и посадку самолётов в каждый момент времени;

3. в случае, если очередь на посадку не пуста, то использовать

взлётную полосу для посадки самолёта, в противном случае

использовать её для взлёта;

4. если какая-либо очередь заполнена, «отправить» самолёт на

другой аэропорт (вывести сообщение в консоль) в случае посадки

или «попросить» подождать некоторое время в случае взлёта;

5. все события (запрос на взлёт/посадку, взлёт/посадка, отказ во

взлёте/посадке) сопровождаются соответствующими

сообщениями в консоли;

6. вести статистику:

сколько получено запросов на взлёт/посадку;   
- сколько принято запросов на взлёт/посадку;

- сколько самолётов взлетело/приземлилось;

- сколько было отказов на взлёт/посадку;

- общее время ожидания взлетевших/приземлившихся самолётов;

- время простоя взлётной полосы (обе очереди пусты).

7. по окончании интервала моделирования (цикл for)вывести в консоль (файл) отчёт, который содержит:

1. общее количество обработанных запросов;

2. кол-во запросов на взлет/посадку;

3. кол-во принятых запросов на взлет/посадку;

4. кол-во отклоненных запросов на взлет/посадку;

5. кол-во взлетевших/приземлившихся самолётов;

6. кол-во самолётов, оставшихся в очереди на взлёт/посадку;

7. время простоя взлётной полосы в процентах;

8. среднее время ожидания для взлёта/приземления;

9. среднее кол-во поступивших запросов на взлёт/приземление.

Для реализации создать классы Runwayи Plane, которые описывают

аэропорт (взлётную полосу) и самолёт соответственно.

Для генерации псевдослучайного числа с законом распределения

Пуассона рекомендуется использовать

...

#include<random>

...

...

Int main(){

...

...

std::random\_device rd; //генератор случайныхчисел.

std::mt19937 gen(rd()); //генератор псевдослучайных чисел, который инициализируется случайным числом rd().

std::poisson\_distribution<>distr(mean); //создание объекта для генерации случайных чисел с распределением Пуассона при заданном среднем (ожидаемом) значении mean.

distr(gen); //генерирование случайного числа с распределением Пуассона с заданным параметром mean.