## Теоретическая часть

* Понятие об итераторах
* Методы и операции итератора
* Прямой и двунаправленный итератор – виды итераторов
* Реализация прямого итератора для односвязного списка
* Реализация прямого итератора для стека

## Практическаячасть

## Задача 1.

Разработайте двунаправленный итератор для двухсвязного списка – шаблонного класса **DblList** из предыдущего задания. Продемонстрируйте работу итератора, на экземпляре двусвязного списка, закрытого типом int.

Операции итератора:

* **begin()**переход итератора на начало контейнера, возвращает 0
* **end()**переход итератора в конец контейнера, возвращает позицию, следующую за последней доступной позицией
* **\***разыменование итератора, возвращает ссылку на значение, хранимое в элементе, на который указывает итератор
* ++ только префиксный инкремент, переход к следующему элементу контейнера
* **--** только префиксный декремент, переход к предыдущему элементу контейнера
* **T get()** получение значения, хранимого в элементе, на который указывает итератор, доступ по чтению
* **void put(T value)** запись нового значения в элемент контейнера, на который указывает итератор, доступ по записи

## Задача 2.

С использованием шаблонного класса стека с итератором проверить правильность расстановки скобок в строке символов.

Cтроки считывать из текстового файла, длина строки не превышает 80 символов (не обрабатывать слишком длинные строки).В файле не менее 3х строк.

Виды скобок: () [] {}

Пример строк:

( тут ( может { быть [](произвольный) текст { например }[ такой]})) - правильно расставлены скобки

({)[()]} – неправильно расставлены скобки

Выводить результаты в консоль – строка для обработки, ниже – строка с точками, соответствующими символам, место обнаружения первой ошибки обозначать символом ^.

## Задача 3.

С использованием шаблонного класса очереди с итератором реализовать очередь нуждающихся в улучшении жилищных условий. Заполняйте очереди из массивов, не менее 10 элементов в массиве.

Элемент очереди должен хранить фамилию, имя и отчество претендента, количество членов семьи, площадь квартиры в м2, количество комнат, дату постановки на учет (день, месяц, год–класс Date). Очередь имеет название, хранит имя бинарного файла для данных.

Реализовать дополнительно операции (при помощи итераторов):

* Поиск в очереди по фамилии – находим первую подходящую фамилию, остальных однофамильцев игнорировать
* Объединение двух очередей в новую очередь. Очередников, находящихся в обеих очередях включать в итоговую очередь только один раз. Имя результирующей очереди должно быть: **Объединение “очередь1” и “очередь2”**.

## Дополнительно

Запись занятия можно скачать [**поэтой ссылке**](https://cloud.mail.ru/public/dXv4/TeEEJw9fX).

Пример чтения данных из текстового файла

