

11. Абстрактные типы данных: очередь.

Теория

Очереди обычно, но не обязательно, упорядочивают элементы по принципу FIFO (первым пришёл — первым ушёл). Исключением являются очереди с приоритетом, которые упорядочивают элементы в соответствии с заданным компаратором или естественным порядком элементов. Независимо от используемого порядка, первым элементом очереди является тот элемент, который будет удалён вызовом `remove()` или `poll()`. В очереди FIFO («первый пришёл – первый ушел») все новые элементы добавляются в конец очереди. В других типах очередей могут использоваться другие правила размещения.

Ознакомиться с реализацией интерфейса `Queue` в официальной документации: <https://docs.oracle.com/en/java/javase/17/docs/api/java.base/java/util/Queue.html>

Задача #1

Реализуйте стек (LIFO — «последний пришёл – первый ушел»), используя не более двух очередей (FIFO — «первый пришёл – первый ушел») и только стандартные методы очереди `add/offer`, `peek`, `poll`, `size`, `isEmpty`. Для проектирования используйте отношение композиция.

В реализованном стеке должны поддерживаться методы **`push`**, **`top`**, **`pop`**, **`empty`** и метод, возвращающий строковое представление всех элементов стека.

Методы класса `StackOnQueue`:

- `void push(int x)` Помещает элемент `x` на вершину стека.
- `int pop()` Удаляет элемент на вершине стека и возвращает его.
- `int top()` Возвращает элемент на вершине стека.
- `boolean empty()` Возвращает `true`, если стек пуст, в ином случае `false`.

Задача #2

Создайте новый тестовый класс. Для проверки работы созданного стека создайте в тестовом классе новый экземпляр класса `StackOnQueue`, добавьте в этот стек два значения, выведите объект, находящийся на вершине стека (без удаления), выведите объект, находящийся на вершине стека и удалите его, проверьте стек на пустоту и выведите информацию о всех элементах стека.