```java

import components.queue.Queue;

/**
 * {@code Queue} represented as a {@code Sequence} of entries, with
 * implementations of primary methods.
 *
 * @param <T>
 *            type of {@code Queue} entries
 * @correspondence this = $this.entries
 */
public class HelloWorld {

    /**
     * Finds {@code x} in {@code q} and, if such exists, moves it to the front
     * of {@code q}.
     *
     * @param <T>
     *            type of {@code Queue} entries
     * @param q
     *            the {@code Queue} to be searched
     * @param x
     *            the entry to be searched for
     * @updates q
     * @ensures <pre>
     * perms(q, #q)  and
     * if <x> is substring of q
     *  then <x> is prefix of q
     * </pre>
     */
    private static <T> void moveToFront(Queue<T> q, T x) {
        boolean included = false;
        int i = 0;
        while (i < q.length()) {
            T temp = q.dequeue();
            if (x != temp) {
                q.enqueue(temp);
            } else {
                included = true;
            }
            i++;
        }
        if (included) {
```

```
44              Queue<T> temp = q.newInstance();
45              temp.enqueue(x);
46              temp.append(q);
47              q.transferFrom(temp);
48          }
49
50      }
51  }
```