Chapter 7: Recursion

EECS 1030

moodle.yorku.ca

Linear search

```
/**
 * Tests whether the given list contains the given
 * element.
 *
 * @param list a list
 * @pre. list != null
 * @param element an element
 * @return true if the given list contains the given
 * element, false otherwise.
 */
public static boolean contains(List<Integer> list,
                               Integer element)
```

Linear search

- Prove that the `contains` method is correct.
- Prove that the `contains` method terminates.
- Give the recurrence relation of the `contains` method.

Linear search

```
/**
 * Tests whether the sublist of the given list
 * starting at the given index contains the given element.
 *
 * @param list a list
 * @pre. list != null
 * @param element an element
 * @param index begin index of the sublist
 * @pre. index <= 0 && index <= list.size()
 * @return true if the sublist of the given list
 * starting at the given index contains the given element,
 * false otherwise.
 */
public static boolean contains(List<Integer> list,
                               Integer element, int index)
```

Linear search

- Prove that the `contains` method is correct.
- Prove that the `contains` method terminates.
- Give the recurrence relation of the `contains` method.

Binary search

```
/**
 * Tests whether the sublist of the given list specified
 * by the given start and end index contains the given elem
 *
 * @param list a list
 * @pre. list != null && list is sorted
 * @param element an element
 * @param begin begin index of the sublist
 * @param end end index of the sublist
 * @pre. begin <= 0 && begin <= end && end <= list.size()
 * @return true if the sublist of the given list
 * specified by the given start and end index contains
 * the given element, false otherwise.
 */
public static boolean contains(List<Integer> list,
                               Integer element,
                               int begin, int end)
```

Binary search

- Prove that the `contains` method is correct.
- Prove that the `contains` method terminates.
- Give the recurrence relation of the `contains` method.

Minimum

```
/**
 * Returns the minimum of the sublist of the given list
 * starting at the given index.
 *
 * @param list a list
 * @pre. list != null && list.size() > 0
 * @param index begin index of the sublist
 * @pre. index <= 0 && index < list.size()
 * @return the minimum of the sublist of the given list
 * starting at the given index.
 */
public static int minimum(List<Integer> list,
                          int index)
```

Minimum

- Prove that the `minimum` method is correct.
- Prove that the `minimum` method terminates.
- Give the recurrence relation of the `minimum` method.

Sum

```
/**
 * Returns the sum of the elements of the sublist of the
 * given list starting at the given index.
 *
 * @param list a list
 * @pre. list != null
 * @param index begin index of the sublist
 * @pre. index <= 0 && index <= list.size()
 * @return the sum of the elements of the sublist of the
 * given list starting at the given index.
 */
public static int sum(List<Integer> list,
                      int index)
```

Sum

- Prove that the `sum` method is correct.
- Prove that the `sum` method terminates.
- Give the recurrence relation of the `sum` method.