

4. (10 points) Give an example of a method that is $O(n)$ for the `ArrayList` and $O(1)$ for the `LinkedList`. Explain why (discuss the internal structure of each container that causes the specific time complexity).

5. (10 points) Give an example of a method that is $O(1)$ for the `ArrayList` and $O(n)$ for the `LinkedList`. Explain why.

6. (10 points) In order to make use of the enhanced for loop (for-each loop), we needed to make the `Roster` class support iterators. Add the needed attributes to the `RosterIterator` class, and complete the implementations of the constructor and `hasNext()` method for the `RosterIterator` class.

```
public class Roster implements List<Student> {
    private Student[] students;
    private int size;
    private int capacity;

    private class RosterIterator implements Iterator<Student> {

    }

    private RosterIterator() {

    }

    public boolean hasNext() {

    }

    }
}
```



7. (15 points) If you were asked to write doubly-linked list that stored **Strings**, what attributes would the class have? What inner class(es) would need to be implemented and what attributes would be would the inner class(es) have, if any? Justify your answer.

8. (25 points) Implement the following methods from the `Roster` class described in question 6.

```
public boolean contains(Student target) {
```

```
}
```

```
public Student indexOf(int index) {
```

```
}
```



```
public boolean add(Student student) {
```

```
}
```