

# CSC 212 Homework # 2 Solution to Selected Problems

## Lists & Queues

### Problem 1

1.

2.

```
public static <T> void moveBack(Listr<T> l, int p) {
    int curToEnd = 0;
    while (!l.last()) {
        curToEnd++;
        l.findNext();
    }

    l.findFirst();
    int headToEnd = 0;
    while (!l.last()) {
        headToEnd++;
        l.findNext();
    }

    l.findFirst();
    int headToP = headToEnd - (p + curToEnd);
    for(int i = 0; i < headToP; i++) {
        l.findNext();
    }
}
```

### Problem 2

1.

```
public void removeK (int k) {
    Node<T> p = current;
    for (int i = 0; i < k; i++) {
        p = p.next;
    }
    current.next = p.next ;
}
```

2.

### Problem 3

1.

```

public int direction(T e) {
    if (current.data.equals(e))
        return 0;

    Node<T> tmp = current.prev;
    while (tmp != null) {
        if (tmp.data.equals(e))
            return -1;
        tmp = tmp.prev;
    }

    tmp = current.next;
    while (tmp != null) {
        if (tmp.data.equals(e))
            return 1;
        tmp = tmp.next;
    }

    return 0;
}

```

2.

### Problem 4

1.

```

public static <T> void remove(Queue<T> q, int[] pos, int k) {
    int l = q.length();
    int j = 0;
    for(int i = 0; i < l; i++) {
        T tmp = q.serve();
        if ((j < k) && (i == pos[j])) {
            j++;
        } else {
            q.enqueue(tmp);
        }
    }
}

```

2.

### Problem 5 [Priority queue]