

# CS251 Homework 1

Leonardo Azopardo

February 14, 2014

## 1 Stacks

- A. Quadratic
- B. Linear
- C. Linear

## 2 Queues

Implementation from the book:

```
public class Josephus {
    public static void main(String[] args) {
        int M = Integer.parseInt(args[0]);
        int N = Integer.parseInt(args[1]);

        // initialize the queue
        Queue<Integer> q = new Queue<Integer>();
        for (int i = 0; i < N; i++)
            q.enqueue(i);

        while (!q.isEmpty()) {
            for (int i = 0; i < M-1; i++)
                q.enqueue(q.dequeue());
            StdOut.print(q.dequeue() + " ");
        }
        StdOut.println();
    }
}
```

### 3 Short Answers

1.

Initialize an array `b` of size `n`, with every element set to zero.

Go through `A` and for every element `i` in `A`, increment the count of `B[i]` by 1.

Go through `B` and print first item with value 2 or more.

2.

Store a pointer `h` to the head node.

Move to the  $N/2 - 1$  node.

Make it point to `h`.

Advance 1 node.

Store a pointer `m` to that node.

Go to the end of the list.

Make the last node point to `m`.

### 4 Analysis of Algorithms

- 1.

- 2.

### 5 Applications of Sorting

Assume `A` contains `N` items and `B` contains `M` items.

Initialize set `C` to store the complement of `A` and `B`

For `i` in `A`

perform binary search for `i` on `B`

if not found:

add `A[i]` to `C`

Do the same for `B`

This would be of the order  $N * M \log M + M * N \log N$ , this may not be the fastest algorithm but it is at least not exponential.