CSE 202 Midterm 2

Olcay Taner YILDIZ

I. Question (Queue) (20 points)

Write the methods

```
void enqueue2nd(Element newElement)
void enqueue2nd(Node newNode)
```

which inserts a new element (node) as a second element for the array (list) implementation. Your method should run in $\mathcal{O}(1)$ time.

II. Question (Queue) (15 points)

Write the method

```
void shrink(int M)
```

which decreases the size of the queue (array implementation) to M so that the new queue can contain at most $M - 1 < N$ elements. You can assume that before shrinking the queue contains at most $M - 1$ elements.

III. Question (Search Trees) (15 points)

Write the method

```
TreeNode randomSearch()
```

which returns a random leaf node from a binary search tree. (Hint: At each node, starting from the root node, go left or right depending on the outcome of a random number)

IV. Question (Search Trees) (15 points)

Write the method

```
int numberOfOddNodes()
```

which returns the number of nodes containing odd numbers in the binary search tree.

V. Question (Hashing) (20 points)

Write the method

```
double loadingFactor()
```

which calculates the loading factor of an hash table. The loading factor of an hash table is the number of elements divided by the size of the hash table. Write the function for both array and link list implementations.

VI. Question (Hashing) (15 points)

A sequence of $n > 0$ integers is called a jolly jumper if the absolute values of the differences between successive elements take on all possible values 1 through n - 1. For instance, 1 4 2 3 is a jolly jumper, because the absolute differences are 3, 2, and 1, respectively. Write the method

```
boolean jollyJumper(int[] sequence)
```

to determine whether a sequence of numbers is a jolly jumper. (Hint: Use an hash table for checking if the absolute differences come from 1 through n - 1.)