SER 222 **Practice** Midterm Exam

Fall 2015, October 7th, 2015

Last Name: _____
First Name: _____
Last 4 digits of ASU ID: _____

Exam Instructions

The exam is open textbook (Algorithms 4e by Sedgewick and Wayne), as well as open note. No electronic items are allowed. Please use a pen (instead of a pencil) if you have one. There are 54 points available and the exam must be completed in 37.5 minutes. This exam has two types of questions:

Short answer questions: There are 18 points of short answer questions. A typical answer is one or two sentences. Each short answer question is worth 1 or 2 points.

Write-in questions: The write-in programming questions are given at the end of the paper. They must be answered on the question paper. There are 6 points of write-in programming questions.

Topic	**Earned**	**Possible**
SA: Data Abstraction		4
SA: Bags, Stacks and Queues		4
SA: Analysis of Algorithms		6
SA: Elementary Sorts, Mergesort		4
Prog: Programming Model		0
Prog: Bags, Stacks and Queues		4
Prog: Analysis of Algorithms		2
Total:		24

1-2 Short Answer: Data Abstraction [4 points]

1. When it comes to functionality (not performance), is it important to know how an ADT is internally implemented? [2 points]

2. Consider the following constructor for an immutable matrix ADT:

```
public class SolnMatrix implements Matrix {
    private final int[][] data;
    public SolnMatrix(int[][] matrix) {
        data = matrix
    }
    // the usual operations follow...
```

Will this class behave as expected? [2 points]

3 Short Answer: Bags, Stacks and Queues [4 points]

3. Trace a queue (called Q) through the following operations:

```
Queue<Integer> Q = new Queue <Integer>();
Q.enqueue(new Integer(3));
Integer X = Q.dequeue();
Q.enqueue(new Integer(7));
Q.enqueue(new Integer(5));
Integer Y = Q.first();
Q.enqueue(new Integer(3));
Q.enqueue(new Integer(9));
Integer Z = Q.dequeue();
```

(a) What is the value of Y after it has been assigned? [2 points]

(b) Give the contents of the queue after the code has been executed. Indicate the elements in order and label the front. [2 points]

4-6 Short Answer: Analysis of Algorithms [6 points]

4. What are the Big-Oh orders of the following growth functions? You should provide a relatively tight upper bound. [2 points]

   $f_1(n) = 100 + 10log_{10}(n)n^2 + 45n$

   $f_2(n) = 100nlog_{10}(n) + nlog_2(n)$

5. What is the Big-Oh order of the following code fragment? The fragment is parametrized on the variable n. Assume that you are measuring the number of println calls. [2 points]

```
for (int i = 1; i <= n; i++)
  for (int j = 1; j <= n; j *= 10)
    System.out.println("Nested loops!");
```

6. Consider the following algorithm that implements linear search:

```
public static boolean find(int target, int[] pool) {
    for(int i = 0; i < pool.length; i++)
        if(pool[i] == target)
            return true;
    return false;
}
```

   If you were asked to analyze the performance of this algorithm, would it be useful to write a growth function that counts the number of times the return statement will execute? That is, use the number of returns as the cost metric. Explain. [2 points]

7-8 Short Answer: Elementary Sorts, Mergesort [4 points]

7. When dealing with systems having low RAM capacity, or analyzing large datasets, space is at a premium. In these cases, algorithms must be designed to reduce their memory foot print. From the sorting algorithm implementations seen in class, which would be the best choice? [2 points]

8. In the lower bound proof for sorting, why must there be at least N! leaves on the decision tree? [2 points]

N/A Programming: Programming Model [0 points]

FYI: if this was the real exam, you would probably be looking at a question asking for fantastic four analysis and code implementation.

9-10 Programming: Bags, Stacks and Queues [4 points]

One fundamental choice in algorithm (or ADT) design is whether to use arrays or linked lists to store information. Both of these data types have advantages, and disadvantages, and choosing the appropriate one can make the different between constant and linear time operations.

For reference, a standard implementation for the nodes of a singly linked list is given below:

```
public class LinearNode<T> {
  private LinearNode<T> next;
  private T element;
  public LinearNode(T elem) { next = null; element = elem; }

  public LinearNode<T> getNext() { return next; }
  public void setNext(LinearNode<T> node) { next = node; }

  public T getElement() { return element; }
  public void setElement(T elem) { element = elem; }
}
```

9. What would be the difference in memory usage for storing a thousand elements in an array vs a linked list? Which takes less space? Explain. [2 points]

10. If you were optimizing for performance and wanted to support potentially adding many new elements to a data structure, would an array or list be more appropriate? Explain. [2 points]

FYI: if this was the real exam, there would probably be a programming question here.

11 Programming: Analysis of Algorithms [2 points]

11. Consider the following method, excerpted from a protein structural prediction algorithm. Assume that any variables not given as parameters are available as globals.

```
//sets initial interaction energy.
//  int n: dimension of square matrix storing protein backbone.
//  double[][] pair: energy matrix.
void energy() {
    double ee = 0;
    //reset all pair interaction energies to zero.
    for(int j = 1; j < n; j++)
        for(int i = 3; i < n-2; i++)
            pair[i][j] = 0.0;

    //<more code follows in actual program>
    //...
```

Give a growth function for *md_fragment* that counts the number of assignments in the inner loop based on n. You may give sigma form. [2 points]