Sample Questions CS10051 Midterm I (From Previous Years)

Note: This is meant to show types of questions asked and approximate level of difficulty

1. Answer True or False
   - F A Repeat-Until loop can execute zero times.
   - F The Sequential Search algorithm uses only sequential operations.

2. Consider the following algorithm. Classify each of the pseudocode statements as S)equential, C)onditional or L)ooping.

<table>
<thead>
<tr>
<th>Step</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>S 1</td>
<td>Get values for NAME, N1,...,N10,000, and T1,...,T10,000</td>
</tr>
<tr>
<td>S 2</td>
<td>Set the value of I to 1 and set the value of Found to NO</td>
</tr>
<tr>
<td>L 3</td>
<td>Repeat steps 4 through 7 until either Found = YES or i &gt; 10,000</td>
</tr>
<tr>
<td>C 4</td>
<td>If NAME is equal to the ith name on the list Ni then</td>
</tr>
<tr>
<td>S 5</td>
<td>Print the telephone number of that person, Ti</td>
</tr>
<tr>
<td>S 6</td>
<td>Set the value of Found to YES</td>
</tr>
<tr>
<td>S 7</td>
<td>Else (NAME is not equal to Ni)</td>
</tr>
<tr>
<td>S 8</td>
<td>Add 1 to the value of I</td>
</tr>
<tr>
<td>C 9</td>
<td>If (Found = NO) then</td>
</tr>
<tr>
<td>S 10</td>
<td>Print the message 'Name is not in the directory'</td>
</tr>
<tr>
<td>S 10</td>
<td>Stop</td>
</tr>
</tbody>
</table>

3. A person is planning to surround a circular garden with black plastic edging. Write an algorithm that inputs the Diameter of the garden in feet and outputs the Length of edging required and its Cost in dollars. Assume edging costs $0.59 per foot and that Pi = 3.1416

   Input: Diameter in feet
   Output: Length of edging in feet – L, Cost of edging
   Other Info: Edging is $.59 per foot
   Formula: Circumference = Pi * Diameter (use to compute length of edging)

   Set the value of Pi to 3.14159
   Get the value of Diameter
   Set the value of Length to value of Pi * Diameter
   Set the value of Cost to value of Length * 0.59
   Output value of Length
   Output value of Cost
   Stop
4. On the grid above draw graphs of the work done for algorithms with each of the following time efficiencies: $\Theta(\log n)$, $\Theta(n^2)$, $\Theta(1)$, $\Theta(n)$, $\Theta(2^n)$

5. The following algorithm is supposed to compute the sum of the squares of the integers 1, 2, 3, …, n. For example, for n=3, it should output the value of $1^2 + 2^2 + 3^2$

Line 1: Input value for n
Line 2: Set the value of SumSq to 0
Line 3: Set the value of i to 1
Line 4: While the value of i $\leq$ n Do
Line 5: Set the value of SumSq to the value of SumSq + i
Line 6: Print Sum
Line 7: Stop

There are 4 errors in this algorithm. Find three. How would you fix each error?

1. Line 5: SumSq + I should be SumSq + I*I
2. Line 5-6: Need statement to increment I to avoid infinite loop
3. Line 5-6: Need an Endwhile
4. Line 6: Print Sum should be Print SumSq
Corrected algorithm
Line 1: Input value for n
Line 2: Set the value of SumSq to 0
Line 3: Set the value of i to 1
Line 4: While the value of i <= n Do
Line 5: Set the value of SumSq to the value of SumSq + i*i
Line 6: Set value of i to value of i+1
Line 7: Endwhile
Line 8: Print SumSq
Line 9: Stop

6. We talked about various algorithm properties in class. **Discuss in detail:**

a. What is required for an algorithm to be Correct?

- Should give answer to correct problem
- Should give correct result for all cases of input
- Should give answer to degree of precision required by problem

b. What is the concept of the Efficiency of an algorithm? How do we measure Efficiency?

It is desirable for an algorithm to be efficient in terms of both memory (or space) utilization and time taken for a given problem.

In order to be efficient in terms of memory an algorithm should use little excess memory beyond that required to hold any initial data. We measure the amount of excess memory used in terms of the amount of input data, N. An algorithm which uses less excess memory for an input of size N is said to be more memory (or space) efficient.

In order to be efficient in terms of time an algorithm should do as little work or take as few steps as possible to solve a problem. We measure time efficiency as the amount of work necessary, or the number of steps performed to solve a problem with a given amount of input data, N. An algorithm which does less work for an input size of N is said to be more time efficient.

We usually express memory (space) efficiency and time efficiency in terms of the amount of excess space required or work done (steps performed) as a function of the amount of input data, N. If the amount of memory used or work done increases linearly as N increases, we say the algorithm is order of magnitude N in terms of space or time. The notation for this is \( \Theta(N) \), pronounced Theta N. An algorithm in which the work increases with the square of N is order of magnitude \( N^2 \) or \( \Theta(N^2) \).
7. Draw flow diagrams to show how the Repeat-Until and While-Do statements work.

8. Answer each part.
   
   a. What is the worst case number of comparisons a sequential search will use to find a name in a list of 160 names?

   \( O(n) \)

   b. What is the worst case order of magnitude time efficiency for a sequential search on a list with \( n \) items?

   8

   c. What is the worst case number of comparisons a binary search will use to find a name in an ordered list of 160 names? \( \log_2(160) \) rounded up

   \( \log_2(160) \)
9.

a. Draw a binary search tree for the following list: 2, 4, 5, 9, 17, 19, 31, 45, 49, 57, 61

```
     19
    /   \
   5     49
  /     /
 2   9   31
 /   /   /  \
4  17  45  57
```

b. If the binary search algorithm is searching for the number 5, how many comparisons will it make? Two