CSCI 6470 Quiz #1 Questions

August 28, 2023 (11:40am-12:00pm EST)

September 28, 2023

Student Name _____ Student ID_____

Rules are as follows. Violation will result in zero credit for the exam/final grade.
   1. Closed book/note/electronics/neighborhood.
   2. Surrender your cell phone to the podium before using the restroom.

There are 4 questions and 40 points in total. Good luck!

1. (**10 points**) Consider the following algorithm `M`. Computer the worse case total number of basic operations for each line of statement and fill it in the corresponding space `[ ]` on the right.

```
Function M(L, n); // L is a list indexed from 1 to n, n>=1
   m = L[1];                          [  1   or ARC (any reasonable constant)]
   k = 2;                             [  1   or ARC ]
   while (L[k] >= m) & (k <= n)       [ 3xn or ARC x n ]
      m = L[k];                       [ 2x(n-1) or ARC x (n-1) ]
      k = k + 1;                      [ 2x(n-1) or ARC x (n-1) ]
   return (m);                        [  1 or ARC ]
```

In addition, the total number of basic operations is ___ARC x n + ARC___ for function `M`.

And what does the algorithm `M` do? Output last number in the monotonically increase sequence from L[1]

2. (**10 points**) This question is about to upper-bound a function.

(1) Let $f(n) = 4n + 2\log_2 n + 2$. Fill in the blank space: $c = 8$ or ARC, $n_0 = 1$ or ARC, such that

$$f(n) \leq cn \qquad \text{when } n \geq n_0$$

(2) Let $f(n) = 4n + 2\log_2 n + 2$. Fill in the blank space $c = 8$ or ARC, $n_0 = 2$ or ARC such that

$$f(n) \leq cn \log_2 n \qquad \text{when } n \geq n_0$$

3. (**10 points**) Mark the following statements either TRUE or FALSE.

(1) [T F] $n^3 + 2n - 6 = O(n^4)$

(2) [T F] $10 \cdot 2^n = O(n^{100})$

(3) [T F] $2^n + 2 \cdot 2^{\frac{n}{2}} = O(2^{\frac{n}{2}})$

(4) [T F] $5 \log_2 n + 5 \ln n = O(\log_{10} n)$

(5) [T F] $7\sqrt{n} = O((\log_2 n)^2)$

4. (**10 points**) Given the following recursive algorithm,

```
function DoSomething(L, n); // L is a list indexed from 1 to n, n>=1
  if (n = 1)
     return (L[1]);
  else
    if (L[n] > L[n/2])
        swap(L[n/2], L[n]);  // assumed n can be evenly divided by 2
    return (DoSomething(L, n-1));
```

(1) Run the algorithm on input list L = [2, 4, 8, 6, 9, 5] and the output of the algorithm is __9__.

(2) DoSomething is called __6__ times on the input L = [2, 4, 8, 6, 9, 5].

(3) What does the algorithm do ? Finding the maximum elements in L.

[The following space will not be graded.]