Quiz Two (CS201)

Your Name: _____ Your SSN: _____

Instructions

- This is a closed-book quiz.
- The quiz has 8 questions, and the full mark is 75.
- Write the answer for each question in the space provided below the question.

1. (15 marks) Give the first 5 terms (k = 1, 2, 3, 4, 5) of the following recursively defined sequences?

   (a). V(k)=V(k-1)+V(k-2) for integers k>=3
   V(1)= 1, V(2)= 1

   Solution: V(3)= 2, V(4)= 3, V(5)= 5

   (b). V(k)=V(k-1)*2 for integers k>=2
   V(1)= 6

   Solution: V(2)=12, V(3)=24, V(4)=48, v(5)=96

   (c). V(k)=V(k+1)-V(k-1) for integers k>=2
   V(1)=1, V(2)=1

   Solution: V(3)=2, V(4)=3, V(5)=5

2. (5 marks) Give the following recursive definition of a set S

   (1). 3 belongs to S (i.e., $3 \in S$).
   (2). for every x, y belongs to S, x+y also belongs to S.

   Which of the following do not belong to S? The answer could be more than 1.

   3, 9, 4, 27, 216, 1345, 1788

   Solution: 4, 1345

3. (5 marks) What does the following algorithm compute? Give a simple formula.

```
int foo(int n , int x ) {
    if (n==1)
      return x;
  else
      return x+foo(n-1,x);
}
```

   Solution: x*n

4. (5 marks) Write the recurrence equation for the following recursive algorithm.

```
int V(int n) {
if (n <= 4)
    return(1);
else
    return(2 * V(n-1));
}
```

Solution:

T(n) = 1, for n <=4

T(n) = 2*T(n-1), for n > 4

5. (5 marks) Which of following statements are true?

(a) $5n^3-2n^2-n+2 = O(n^3)$

(b) $100n^2 + n^3-n+2 = O(n^2)$

(c) $5n^3-1000n^{200}-2^n+2 = O(n^{200})$

(d) $5n^3-1000n^{200}-2^n+2 = O(2^n)$

(e) $3n^2 log n + 200n + (n+100)^2 = O(n^2 log n)$

(f) $3n^2 log n + 200n + (n+100)^2 = O((n+100)^2)$

Solution: (a)(d)(e) are true, (b)(c)(f) are false.

6. (5 marks) Rank the following typical bounds in increasing order of growth rate:

$O(log\ n)$, $O(n^3)$, $O(3^n)$, $O(n)$, $O(nlogn)$, $O(n^2)$

Solution: O(log n), O(n), O(nlogn), $O(n^2)$, $O(n^3)$, $O(3^n)$

7. (20 marks) For each of the following loops, give the tightest upper bound using big O notation.

(1)
```
for ( int i = 0; i < n; i++)
{       sum ++;
        for ( int j = 0; j < n; j++)
            sum ++;
}
```

Solution: $O(n^2)$

(2)
```
for ( int i = 0; i < n; i++)
    for ( int j = 0; j < i; j ++)
        sum++;
```

Solution: $O(n^2)$

(3)
```
for ( int i = 0; i < n; i++)
{
    for ( int j = 0; j < i; j++)
        sum++;
    for ( int j = 0; j < n*n; j++)
        sum++;
 }
```

Solution: O(n^3)

(4)
```
for ( int i = 0; i < n; i ++)
    for ( int j = 1; j < n*n; 2*j)
        sum++;
```

Solution: O(n*log*n)

8. (15 marks) An algorithm takes 1 ms for input size N = 100. How long will it take for input size of 500 if the running time is the following?

(a). linear

Solution: 500/100 * 1 = 5 ms

(b). quadratic

Solution: $(500/100)^2$*1 = 25 ms

(c). cubic

Solution: $(500/100)^3$*1 = 125 ms