CS204 Discrete Mathematics, Spring 2018

Recitation #5

Time: 2018.04.05 (Thu) 19:00 ~ 19:30

TA: Donghyun Lim

1. A palindrome is a string that reads the same forward and backward. Describe an algorithm in pseudo code for determining whether a given string is a palindrome

Sol)

```
is_palindrome(s:string):
    i := leftmost index of s
    j := rightmost index of s
    while i<j:
        if s[i] != s[j]:
            return False
        i := i+1
        j := j-1
    return True
```

2. Recall the mathematical definitions of big-O, big-Omega, and big-Theta notations.

Sol) Omitted

3. Determine without proof whether x^3 is O(g(x)) for each of these functions g(x).

a) g(x) = x^2

b) g(x) = x^3

c) g(x) = $x^2 + x^3$

d) g(x) = $x^2 + x^4$

e) g(x) = 3^x

f) g(x) = $x^3 / 2$

Sol) F, T, T, T, T, T

4. Prove that n! is O(n^n).

Sol)

For all x = 1, 2, … n, we have x <= n. By multiplying all n inequalities, we have n! <= n^n.

5. Give a big-O estimate for the number additions used in the following code. Ignore possible implicit additions used for incrementing the index variable i and j in the for-loops.

```
t := 0
for i := 1 to n
    for j := 1 to n
        t := t + i + j
```

Sol) O(n^2)

6. Fibonacci number with initial constants f_1=1,f_2=1 can be calculated by the two naïve ways; one is to use a recursive algorithm and the other is to use an iterative algorithm; e.g., see the below description:

```
Rec_fibo(n):
    If n equals 1 or 2: return 1;
    Else: return Rec_fibo(n-1) + Rec_fibo(n-2);

Iter_fibo(n):
a=0, b=1, c=1;
for k=2 to n:
    c = a+b;
    a = b;
    b = c;
end for;
return b;
```

Describe pros and cons of each approach.

Sol) Recursive version might be shorter than iterative version since it just represents the structure of the solution. But it may takes quite more cost. On the other hand, iterative version might be harder to implement, but it can reduce time complexity more than recursive version does.