

GIT Department of Computer Engineering

CSE 222/505 - Spring 2018

Homework 02

Due date: March 19 2018 – 12:00

1. Prove using only the definitions of asymptotic notations. (15p)

- a. $n2^n$ is in $O(2^{2n})$
- b. $n!$ is in $\Omega(2^n)$
- c. $(\log n)$ is in $\Theta(\log_{64} n)$

2. Sort the following functions from slowest to fastest in terms of their growth. Using limit estimation. (10p)

$(\lg n)^{\lg n}$, $\lg(n!)$, n , $\lg(\lg(n))$, $\lg(2^n)$, $\lg(\lg(\sqrt{n}))$

3. Solve the following recurrence relation using the substitution method. (15p)

$$T(n) = 2T(n-1) + 1 \quad n > 1, \quad T(n) = 1 \quad n = 1$$

4. Explain the running time of $f(n)$ using recurrence relation. (10p)

f(n):

```
if (n == 1)
    return 1
else
    return f(n-1) + f(n-1)
```

5. What does do UnknownFunction? Analyze the running time of UnknownFunction using proper asymptotic notations. (15p)

UnknownFunction(n):

```
i = 0
while (n%2 == 0)
    n = n/2
    i++
return i
```

6. Write Insertion sort with pseudocode Explain analyze of your algorithm worst-case, best-case and average-case using proper asymptotic notations. (20p)
7. Calculate the running time of the Test function. Show your calculation in proper asymptotic notations. (20p)

Test(n):

```
for (int i = 1 ; i < 2n ; i+=4i)
    for (int j = n ; j > 0; j--)
        if ( i*j == target)
            target = checkFunc(n);
        else
            print();
```

checkFunc(n){

```
foo(n);
if (n == 1)
    return 1;
else
    return checkFunc(n/2);
}
```

foo(n){

```
for (int i = 0 ; i < n ; ++i)
    print();
}
```

*Assume that arithmetic operations and print() can be done in constant time.

Note:

- Do not email your homework or submit it through moodle.
- Your submissions will be handwritten.
- You should handover the submissions to the Arzu Kakışım before 12:00 on due date.
- Fatma Nur Esirci will score this homework.