

Home Work

CS221: Data Structures and Algorithms

Usman Institute of Technology

Fall 2019

Running Time and Asymptotic Analysis

1. Calculate the running time of the following:

A.

```
for(i = 1; i <= n*n; i=i+1)
    for (j = 1; j <= i; j++)
        print("Hello");
```

B.

```
for(i = 1; i <= n*n; i=i+1)
    for (j = 1; j <= i; j++)
        if( j == 100)
            break;
```

C.

```
for(i = 1; i <= n; i=i*2)
    for (j = 1; j <= i; j++)
        print("Hello Again");
```

2. Write an algorithm to sum the all integers in the given array. Calculate the running time of the algorithm as well.
3. What do you understand by *rate of growth* or *order of growth*? Why it is important in analysis of an algorithm?
4. Calculate $T(n)$ for Selection Sort algorithm. Also analyze the best and worst cases for Selection Sort.
5. Calculate Ω (lower bound) for different variations of Bubble Sort algorithms as discussed in the class.
6. Rank the following functions according to their growth rate, from slowest growing to fastest growing:
- $8n^3$
 - $n \log n$
 - 64
 - $\log n$
 - $5n$
 - 2^n
 - n^2
7. Prove or disprove the following for Big-O:
- $f(n) = n + 30, g(n) = n^2 + 3n$
 - $f(n) = n^2 + 2n - 10, g(n) = n^4$
 - $f(n) = n^3 * 3n, g(n) = 3n - 1$
8. For each of the following pairs of functions, either $f(n)$ is $O(g(n))$, $f(n)$ is $\Omega(g(n))$, or $f(n) = \Theta(g(n))$. Determine which relationship is correct.
- $f(n) = \log n^2; g(n) = \log n + 5$
 - $f(n) = n; g(n) = \log n^2$
 - $f(n) = \log \log n; g(n) = \log n$
 - $f(n) = n; g(n) = \log^2 n$
 - $f(n) = n \log n + n; g(n) = \log n$
 - $f(n) = 10; g(n) = \log 10$
 - $f(n) = 2^n; g(n) = 10n^2$
 - $f(n) = 2^n; g(n) = 3^n$