

EECE.3220: Data Structures

Spring 2019

Key Questions

Algorithmic Complexity (Lectures 7 & 8)

QUESTIONS

1. Describe how to analyze the worst-case execution time of an algorithm.
2. Explain big O notation.
3. Describe a general linear search algorithm for finding a value in an array, including an analysis of its worst-case execution time.
4. Describe a general binary search algorithm for finding a value in an array, including an analysis of its worst-case execution time.
5. Describe a general selection sort algorithm for ordering the values of an array, including an analysis of its worst-case execution time.

EXAMPLES

1. Determine the worst-case execution time, $T(n)$, of each function listed below as a function of n , and express that execution time using big O notation ($T(n) = O(?)$).

a.

```
int F(int n) {  
    int i, res;  
1    if (n < 2)  
  
2        return 1;  
  
3    else {  
4        res = 1;  
5        for (i = 2; i <= n; i++)  
6            res *= i;  
7        return res;  
    }  
}
```

b.

```
unsigned F(unsigned n) {  
1    unsigned res = 0;  
  
2    for (i=0; i<n+1; i++)  
3        for (j=0; j<n+1; j++)  
4            res = res + j;  
5    return res;  
}
```