Homework 2 COM S 311

Alec Meyer

September 9, 2020

Question 1

a.

$2^{2n} \in O(2^n)$

Claim: $2^{2n} \notin O(2^n)$

$f(n) \leq c \times g(n)$

f(n) = 2^{2n}, g(n) = 2^n

plug in f(n) and g(n)
$2^{2n} \leq c \times 2^n$
$\frac{2^{2n}}{2^n} \leq c$
$2^n \leq c$

Here we have a constant 'c' being greater than or equal to a function of n. This is not possible as 2^n will surpass the value of c. Therefore, since there is no constant value c to satisfy this claim we know that this statement is false.

$\therefore 2^{2n} \notin O(2^n)$ b.

Question 2

a.

Inner:

$$\sum_{k=1}^{j} c_1 = c_1 j$$

Middle:

$$\sum_{j=i}^{n} (c_2 + c_1 j)$$

$$\sum_{j=i}^{n} c_2 + \sum_{j=i}^{n} c_1 j$$

$$\sum_{j=i}^{n} c_1 j = c_1 (\frac{n(n+1)}{2} - \frac{i(i-1)}{2}$$

$$\sum_{j=i}^{n} c_2 = c_2(n-i+1)$$

Outer:

$$\sum_{i=1}^{n-1} [c_2(n-i+1) + c_1(\frac{n(n+1)}{2} - \frac{i(i-1)}{2})]$$

$$\sum_{i=1}^{n-1} [c_2 n - c_2 i + c_2 + \frac{c_1 n^2}{2} + \frac{c_1 n}{2} - \frac{c_1 i^2}{2} - \frac{c_1 i}{2}]$$

$$= c_2 n^2 - \frac{c_2 n(n+1)}{2} - 1 + c_2 + \frac{c_1 n^3}{2} + \frac{c_1 n^2}{2} - \frac{n^2(n+1)}{2} - 1 - \frac{n(n+1)}{2} - 1$$

$$\in O(n^3)$$

Question 3

a.

```
for(int i = 0; i < A.size; i++)
        if(binarySearch(T-A[i]))
                TRUE
FALSE
```

FOR loop time:

$$\sum_{i=0}^{n} BinarySearch$$

We know that Binary Search is O(logn)

We Binary Search n times, therefore the runtime of this algorithm is O(nlogn)