CST 370 – Homework 3
Ivan Alejandre, ID: 8889
01/20/2020

1.a) Q(1) = 1

b) Q(2) = 4

c) Q(3) = 9

d) It appears that Q(n) computes the square of n.

2a) M(n) = 2*M(n-1); M(1)=3

M(n-1) = 2*M(n-2)
M(n-2) = 2*M(n-3)
M(n-3) = 2*M(n-4)

M(n) = 2*2*2*2*M(n-4) → M(n) = 2*k*M(n-k)

find k: n-k = 1 → k = n-1, plug back into equation above

M(n) = 2*(n-1)*M(n-(n-1))
= 2*n – 2 * 3
**= O(n)**

b) M(n) = 3*M(n-1); M(1) = 4 (SAME AS PART A BUT WITH DIFFERENT CONSTANTS)

M(n-1) = 3*M(n-2)
M(n-2) = 3*M(n-3)
M(n-3) = 3*M(n-4)

M(n) = 3*3*3*3*M(n-4) → M(n) = 3*k*M(n-k)

find k: n-k = 1 → k = n-1, plug back into equation above

M(n) = 2*(n-1)*M(n-(n-1))
= 2*n – 2 * 4
**= O(n)**

3a) This algorithm finds the minimum value of the array.

b) M(n) = M(n-1) + 1

c) M(1) = 1

d) M(n-1) = M(n-2) + 1 + 1
M(n-2) = M(n-3) + 1 + 1 + 1
M(n-3) = M(n-4) + 1 + 1 + 1 + 1
M(n) = M(n-k) + k
k = n-1
M(n) = 1 + n – 1 = n

**O(n)** time complexity

4a) The sum of n cubes

b) M(n) = M(n-1) + 2

c) M(1) = 0

d) M(n-1) = M(n-2) + 2 + 2
M(n-2) = M(n-3) + 2 + 2 + 2
M(n-3) = M(n-4) + 2 + 2 + 2 + 2
M(n) = M(n-k) + 2*k
k = n-1
M(n) = 0 + 2*(n-1) = 2*n – 2

**O(n)** time complexity

5) Start with two indices, one for each array. If both elements are the same, output to console and increment by 1. If not, compare the values of the elements and increment the index of the smaller element. Repeat this step until a match is found, if not, increment the other index and repeat.