

# Question 3

3. Analyze the big-oh run time for each function with respect to parameter n:

```
void A(int n) // O(1)
{
    if(n < 10) time++; // O(1)
    else {
        for(int i = 0; i < n; i++) // O(n) = O(n) * 1
            time++;
        A(n-2); → T(n-2)
    }
}
```

$O(n^2)$

$T(n) = T(n-2) + cn$  → got from 100P

$$T(n) \quad \begin{array}{c} n \\ / \\ c(n-2i) \end{array} \quad \begin{array}{l} \text{work done} \\ i=0 \end{array} \quad \begin{array}{l} i \\ = \\ c(n-2i) \end{array} \quad \begin{array}{l} i \text{ represents} \\ \text{the level} \\ \text{we're on} \end{array}$$

$$T(n-2) = T(n-4) + c(n-2) \quad \begin{array}{c} c(n-2) \\ / \\ c(n-4) \end{array} \quad \begin{array}{l} i=1 \\ = \\ c(n-2i) \end{array}$$

$$T(n-4) = T(n-6) + c(n-4) \quad \begin{array}{c} c(n-4) \\ / \\ c(n-6) \end{array} \quad \begin{array}{l} i=2 \\ = \\ c(n-2i) \end{array}$$

⋮

$$T(n-2i) = 1 \quad 1 \text{ // base case}$$

How many levels?

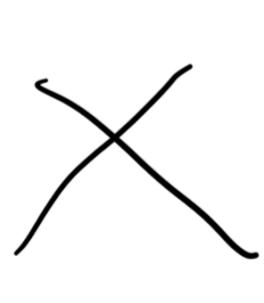
$$n - 2i = 1$$

$$2i = 1 + n$$

$$i = \frac{1+n}{2}$$

Solve the runtime

Work per level



length of tree

$$\rightarrow n * \frac{n}{2} = \frac{1}{2} n^2$$

$O(n^2)$

Proof w/ summation:

$$\sum_{i=0}^{n/2} c(n-2i) = c [n + (n-2) + (n-4) \dots 1]$$

$n/2$