

# DESIGN AND ANALYSIS OF ALGORITHMS

ICS2301

David Kamau Wambui

SCT222-0289/2022

## Assignment.

Find the time complexity of Quick Sort assuming the array is already sorted.

Given array:

[5, 10, 15, 20, 25, 30]

## Solution

- If the first element is chosen as pivot
- The array is already sorted
- One portion has  $n-1$ , the other has 0

Therefore:

$$\begin{aligned}T(n) &= T(n-1) + c(n) \\&= [T(n-2) + c(n-1)] + c(n) \\&= T(n-2) + c(n-1) + c(n) \\&= [T(n-3) + c(n-2)] + c(n-1) + c(n) \\&= T(1) + C(2+3+4+\dots+n)\end{aligned}$$

$$\text{Summation of } i = n(n+1)/2 \Rightarrow (n^2 + n) * 1/2$$

Ignore the constants and pick the highest power of  $n$  which is  $n^2$

$\Rightarrow O(n^2)$  will be the time complexity. Ans  $\Rightarrow T(n)=O(n^2)$

## Space Complexity:

- Recursive calls up to depth  $n$   
 $\Rightarrow O(n)$

## Design Strategy

$\Rightarrow$  Divide and Conquer with Balanced Partitioning