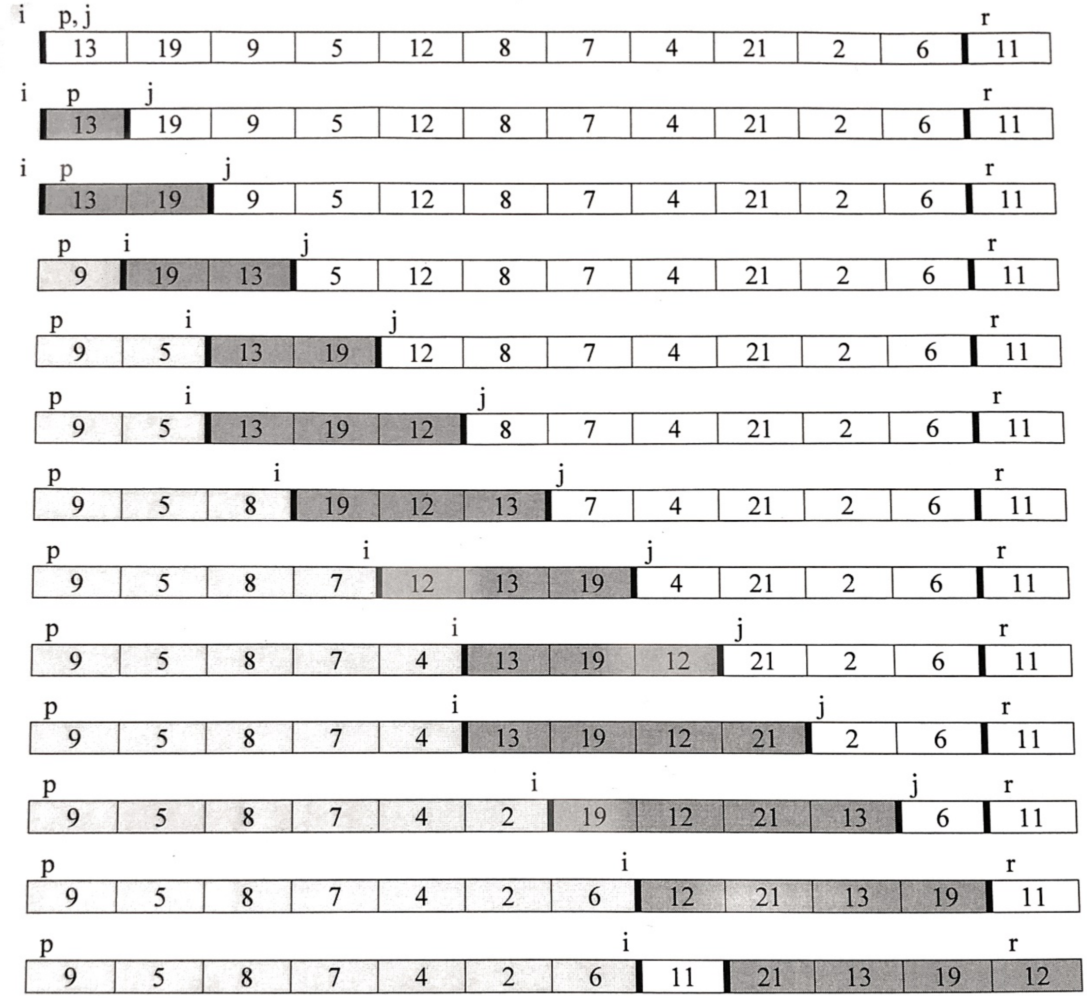
1. By Hoang Do

1. A = {13, 19, 9, 5, 12, 8, 7, 21, 2, 6, 11}



2. By Duyen Tran

the values of q that each partition returns is an alternation between the smallest value and the biggest value

Quicksort(A, p, q-1)'s q value is the first index of it's subarray

QuickSort(A, q+1, r)'s q value is the last index of it's subarray

This is worst case because it's already in an order

3. By Bonnie Liu

QUICKSORT(A, p, r)

```
if p < r
    q = PARTITION(A, p, r)
    QUICKSORT(A, p, q - 1)
    QUICKSORT(A, q + 1, r)
```

So.

$$T(n) = T(0) + T(n-1) + \Theta(n)$$
$$= T(n-1) + \Theta(n)$$
$$= T(n-1) + cn \quad (c: \text{postive constan})$$

Use recursion tree method:

cn — $T(n-1)$

cn — $c(n-1)$ — $T(n-2)$

cn — $c(n-1)$ — $c(n-2)$ — ⋮ — c

$$T(n) = cn + c(n-1) + c(n-2) \dots + c$$
$$= c(n + (n-1) + \dots + 1)$$
$$= c\left(\frac{n(n+1)}{2}\right)$$
$$= \Theta(n^2)$$

i=1

4. By Dangnhi Ngo

Upper bound: $T(n) \leqslant T(n-1) + cn$ (c: positive constant)

Guess: $T(n) = O(n^2)$

$\leqslant dn^2$ (d: positive constant)

$T(n-1) \leqslant d(n-1)^2$

$T(n-1) \leqslant d(n^2 - 2n + 1)$

Substitution: $T(n) \leqslant dn^2 - 2dn + d + cn$

$= dn^2 + (c-2d)n + d$

$\leqslant dn^2$ (if $(c-2d)n + d \leqslant 0$)

$\Rightarrow c - 2d \leqslant 0$

$\Rightarrow d \geqslant c/2$

$T(n) = O(n^2)$ (1)

Lower bound: $T(n) \geqslant T(n-1) + cn$ (c: positive constant)

Guess: $T(n) = \Omega(n^2)$

$\geqslant d(n^2)$ (d: positive constant)

$T(n-1) \geqslant d(n-1)^2$

$T(n-1) \geqslant d(n^2 - 2n + 1)$

Substitution: $T(n) \geqslant dn^2 - 2dn + d + cn$

$= dn^2 + (c-2d)n + d$

$\geqslant dn^2$ (if $(c-2d)n + d \geqslant 0$)

$\Rightarrow c - 2d \geqslant 0$

$\Rightarrow d \leqslant c/2$

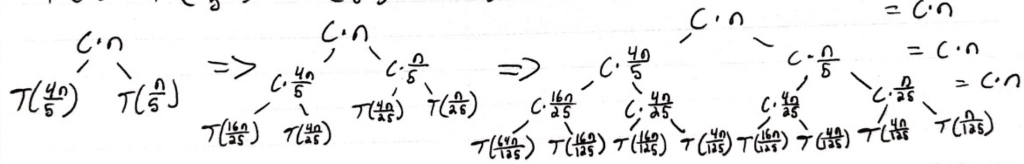
$T(n) = \Omega(n^2)$ (2)

From (1) and (2),

$T(n) = \Theta(n^2)$

5. By David Baumann

5 $T(n) = T(\frac{4n}{5}) + T(\frac{n}{5}) + \Theta(n)$



$\Omega(n \log_5 n) \leq T(n) \leq O(n \log_{4/5} n)$

$\therefore T(n) = \Theta(n \lg n)$

6.By Karamel Quitayen

Minimum depth is repeatedly taking the smaller subproblem of the two, the branch that is proportional to α.

$$1 = \alpha^k n$$
$$k = \log_\alpha \frac{1}{n}$$
$$k = -\frac{\lg n}{\lg \alpha}$$

Similarly, maximum depth is repeatedly taking the larger of the two subproblems, and this branch is proportional to $1 - \alpha$.

$$1 = (1-\alpha)^k n$$
$$k = \log_{(1-\alpha)} \frac{1}{n}$$
$$k = -\frac{\lg n}{\lg(1-\alpha)}$$