

## Group Problem Set #9

Given the array  $a = [40\ 10\ 30\ 70\ 50\ 60\ 20]$  the following depicts the array after each partition of the selection algorithm to find the 4<sup>th</sup> largest value. The element at index  $p$  is used as the pivot element.

$a = [40\ 10\ 30\ 70\ 50\ 60\ 20]$

$\text{selection}(a, 0, 6, 4)$

$p = 0, r = 6, i = 4$

partition around  $a[0] = 40, q = 2: \quad a = [20\ 10\ 30\ 70\ 50\ 60\ 40]$

$k = q - p + 1 = 2 - 0 + 1 = 3$

since  $i > k$ ,  $\text{selection}(a, q+1, r, i-k) = \text{selection}(a, 3, 6, 1)$

$p = 3, r = 6, i = 1$

partition around  $a[3] = 70, q = 5: \quad a = [20\ 10\ 30\ 40\ 50\ 60\ 70]$

$k = q - p + 1 = 5 - 3 + 1 = 3$

since  $i \leq k$ ,  $\text{selection}(a, p, q, i) = \text{selection}(a, 3, 5, 1)$

$p = 3, r = 5, i = 1$

partition around  $a[3] = 40, q = 3: \quad a = [20\ 10\ 30\ 40\ 50\ 60\ 70]$

$k = q - p + 1 = 3 - 3 + 1 = 1$

since  $i \leq k$ ,  $\text{selection}(a, p, q, i) = \text{selection}(a, 3, 3, 1)$

$p = 3, r = 3, i = 1$

Since  $3 == 3$  return  $a[3] = 40$

Perform the same sequence of steps except run it on the quicksort algorithm instead of the selection algorithm.

```
def quickSort(alist, p, r):
    if p >= r:
        return
    q = partition(alist, p, r)
    quickSort(alist, p, q)
    quickSort(alist, q+1, r)
```