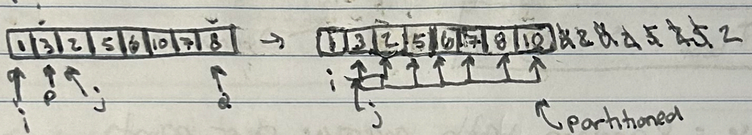
Adam Caudle Hw 5

1) Write pseudo-code for partition(A, p, q)

```
def partition(A, p, q)          <- segment of array to partition
    pivot = A[q]                # chose pivot
    i = p-1                     # tracking variables   j = mover (middle ptr)
    j = p                                              i = back mover/tracker
    for( j=p ; j <= pivot ; j++ ){        . moving var touch q
        if ( A[j] < pivot )      # if mover finds index < pivot move
            i++                    back tracker and swap with mover
            swap(A[i], A[j]) }
    Swap ( A[i+1], A[q] )   # move
    return i+1
```

$j = 0, 1, 2, 3, 4, 5, 6$



2) Consider insertsort. Suppose A 1% prob to be monotonically decreasing. Show that in this case the T_{avg} is $O(n^2)$.

Insertsort $(A[1..n], i)$ — Prob
- monotonically = Worst case $= O(n^2) = 1\%$
- random · avg case $= O(n^2) = 99\%$

① $T_{avg}(n) = \frac{1}{n}\sum_{r=1}^{n} \frac{1}{100} O(n^2) + \frac{99}{100} O(n^2)$

② I.H: $\boxed{\exists i > 0 \text{ s.t } \forall n,\ T_{avg}(i) \leq c i^2}$

③ $T_{avg}(n) = \frac{1}{n}\sum_{r=1}^{n} .01(an^2) + .99(an^2)$ ✓

$= \frac{1}{n}\sum_{r=1}^{n} .01(an^2) + \frac{1}{n}\sum_{r=1}^{n} .99\, an^2$ ✓

$= \frac{1}{n}\sum_{r=1}^{n} an^2$ ✓ $(n = x)$

$= \frac{a}{n}\int_1^n x^2\, dx$

$= \frac{a}{3n} x^3 + c$ ✓

$= \frac{a}{3n} n^3 + c = \frac{a}{3} n^2$ ✓

$\boxed{\frac{a}{3}n^2 + c \leq Cn^2 \text{ when } C >> a}$

* Because avg case and worst case complexities (monotonically = worst case) the average time complexity is unchanged by the worst-case condition and thus it is impossible for the complexity to be anything other than $O(n^2)$.