Find the median using Selection (DAC) where v = 2nd element.

A = [7, 8, 11, 3, 9, 1, 4, 10, 3, 8, 6, 12]

select(array [7, 8, 11, 3, 9, 1, 4, 10, 3, 8, 6, 12], K = 6):

split_array([7, **8**, 11, 3, 9, 1, 4, 10, 3, 8, 6, 12]):

A_L = [7, 3, 1, 4, 3, 6]; N_L = 6

A_P = [8, 8]; N_P = 2

A_R = [11, 9, 10, 12]; N_R = 4

Since $K \leq N_L$ => 6 = 6, return select(A_L, K)

select([7, 3, 1, 4, 3, 6], K = 6):

split_array([7, **3**, 1, 4, 3, 6]):

A_L = [1]; N_L = 1

A_P = [3, 3]; N_P = 2

A_R = [7, 4, 6]; N_R = 3

Since K > (N_L + N_P) => 6 > (1 + 2), return select(A_R, K = 6 – (1 + 2) = 3)

select([7, **4**, 6], K = 3):

split_array([7, **4**, 6]):

A_L = [0]; N_L = 0

A_P = [4]; N_P = 1

A_R = [7, 6]; N_R = 2

Since K > (N_L + N_P), return select(A_R, K – (N_L + N_P) => K = 3 – 1 = 2)

select([7,6], K = 2):

split_array([7, **6**]):

A_L = [0]; N_L = 0

A_P = [6]; N_P = 1

A_R = [7]; N_R = 1

Since K > (N_L + N_P): return select(A_R, K – (N_L + N_P) => K = 2 – 1 = 1)

select([7], K = 1):

split_array([7]):

A_L = [ ]; N_L = 0

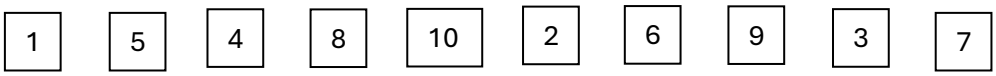
A_P = [7]; N_P = 1

A_R = [ ]; N_R = 0

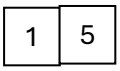
Since $N_L < K \leq (N_L + N_P)$ => $0 < 1 \leq 1$, return $A_P[0]$ = **7**

Count and list (in ascending order) the inversions in A = [1, 5, 4, 8, 10, 2, 6, 9, 3, 7] using merge sort.

After merge sort (split array into halves until only one element is left):



Merge:

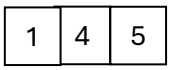


Inversions:

Inversions count = 0

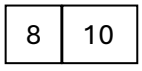
Inversion:

Inversions count = 0 + 0



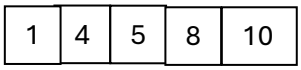
Inversions: (5, 4)

Inversions count = 0 + 0 + 1



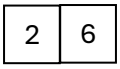
Inversions: (5, 4)

Inversions count = 0 + 0 + 1 + 0



Inversions: (5, 4)

Inversions count = 0 + 0 + 1 + 0 + 0

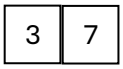


Inversions: (5, 4)

Inversions count = 0 + 0 + 1 + 0 + 0 + 0

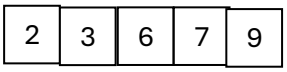
Inversions: (5, 4)

Inversions count = 0 + 0 + 1 + 0 + 0 + 0 + 0



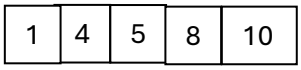
Inversions: (5, 4)

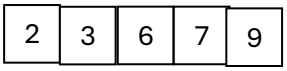
Inversions count = 0 + 0 + 1 + 0 + 0 + 0 + 0 + 0

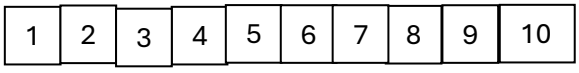


Inversions: (5, 4), (6, 3), (9, 3), (9, 7)

Inversions count = 0 + 0 + 1 + 0 + 0 + 0 + 0 + 0 + 3







Inversions: **(5, 4), (6, 3), (9, 3), (9, 7), (4, 2), (5, 2), (8, 2), (10, 2), (4, 3), (5, 3), (8, 3), (10, 3), (8, 6), (10, 6), (8, 7), (10, 7), (10, 9)**

Inversions count = 0 + 0 + 1 + 0 + 0 + 0 + 0 + 0 + 3 + 13 = **17**