**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]):

AL = [7, 3, 1, 4, 3, 6]; NL = 6

AP = [8, 8]; NP = 2

AR = [11, 9, 10, 12]; NR = 4

Since K ≤ NL => 6 = 6, return select(AL, K)

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

split\_array([7, **3**, 1, 4, 3, 6]):

AL = [1]; NL = 1

AP = [3, 3]; NP = 2

AR = [7, 4, 6]; NR = 3

Since K > (NL + NP) => 6 > (1 + 2), return select(AR, K = 6 – (1 + 2) = 3)

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

split\_array([7, **4**, 6]):

AL = [0]; NL = 0

AP = [4]; NP = 1

AR = [7, 6]; NR = 2

Since K > (NL + NP), return select(AR, K – (NL + NP) => K = 3 – 1 = 2)

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

split\_array([7, **6**]):

AL = [0]; NL = 0

AP = [6]; NP = 1

AR = [7]; NR = 1

Since K > (NL + NP): return select(AR, K – (NL + NP) => K = 2 – 1 = 1)

select([7], K = 1):

split\_array([7]):

AL = [ ]; NL = 0

AP = [7]; NP = 1

AR = [ ]; NR = 0

Since NL < K ≤ (NL + NP) => 0 < 1 ≤ 1, return AP[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):

7

3

9

6

2

10

8

4

5

1

Merge:

5

1

Inversions:

Inversions count = 0

4

Inversion:

Inversions count = 0 + 0

5

4

1

Inversions: (5, 4)

Inversions count = 0 + 0 + 1

10

8

Inversions: (5, 4)

Inversions count = 0 + 0 + 1 + 0

10

8

5

4

1CVC

Inversions: (5, 4)

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

6

2

Inversions: (5, 4)

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

9

6

2

Inversions: (5, 4)

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

7

3

Inversions: (5, 4)

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

7

9

6

3

2

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

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

10

8

5

4

1CVC

7

9

6

3

2

3

10

9

8

7

6

5

4

2

1CVC

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**