```
def pClosest(points, K):

        points.sort(key = lambda K: K[0]**2 + K[1]**2)
        return points[:K]
points = [[1,3],[-2,2],[5,8],[0,1]]
K = 2
print(pClosest(points, K))
```

```

File Edit Format Run Options Window Help

```python
class Solution(object):
    def fourSumCount(self, A, B, C, D):
        sums =[}
        for i in A:
            for j in B:
                if i+j not in sums:
                    sums[i+j] = 1
                else:
                    sums[i+j] +=1
        counter = 0
        for i in C:
            for j in D:
                if -1 * (i+j) in sums:
                    #print(-1 * (i+j))
                    counter+=sums[-1*(i+j)]
        return counter
ob1 = Solution()
print(ob1.fourSumCount([1, 2], [-2, -1],  [-1, 2],  [0, 2]))
```

IDLE Shell 3.12.4

File Edit Shell Debug Options Window Help

```
Python 3.12.4 (tags/v3.12.4:8e8a4ba, Jun  6 2024, 19:30:16) [MSC v.1940 64 bit (
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: D:/DAA/kth pair XY plane.py
[[3, 3], [-2, 4]]
>>>
==================== RESTART: D:/DAA/kth pair XY plane.py ====================
[[0, 1], [-2, 2]]
>>>
==================== RESTART: D:/DAA/compute tuple.py ====================
Traceback (most recent call last):
  File "D:/DAA/compute tuple.py", line 1, in <module>
    class Solution:
  File "D:/DAA/compute tuple.py", line 2, in Solution
    def fourSumCount(self, A: List[int], B: List[int], C: List[int], D: List[int
]) -> int:
NameError: name 'List' is not defined. Did you mean: 'list'?
>>>
==================== RESTART: D:/DAA/compute tuple.py ====================
2
>>>
==================== RESTART: D:/DAA/compute tuple.py ====================
1
>>>
==================== RESTART: D:/DAA/compute tuple.py ====================
2
>>>
```

File Edit Format Run Options Window Help

```python
        else:
            medOfMed = kthSmallest(median, 0, i - 1, i // 2)

        pos = partition(arr, l, r, medOfMed)

        if pos - l == k - 1:
            return arr[pos]
        if pos - l > k - 1:
            return kthSmallest(arr, l, pos - 1, k)

        return kthSmallest(arr, pos + 1, r, k - pos + l - 1)

    return 999999999999

def swap(arr, a, b):
    temp = arr[a]
    arr[a] = arr[b]
    arr[b] = temp

def partition(arr, l, r, x):
    for i in range(l, r):
        if arr[i] == x:
            swap(arr, r, i)
            break

    x = arr[r]
    i = l
    for j in range(l, r):
        if arr[j] <= x:
            swap(arr, i, j)
            i += 1
    swap(arr, i, r)
    return i

def findMedian(arr, l, n):
    lis = []
    for i in range(l, l + n):
        lis.append(arr[i])
    lis.sort()
    return lis[n // 2]

if __name__ == '__main__':
    arr = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
    n = len(arr)
    k = 6
    print("K'th smallest element is", kthSmallest(arr, 0, n - 1, k))
```

IDLE Shell 3.12.4

File Edit Shell Debug Options Window Help

```
Python 3.12.4 (tags/v3.12.4:8e8a4ba, Jun  6 2024, 19:30:16) [MSC v.1940 64 bit (
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
================== RESTART: D:/DAA/kth smallest elelment.py ==================
K'th smallest element is 5
>>>
================== RESTART: D:/DAA/kth smallest elelment.py ==================
K'th smallest element is 6
>>>
```