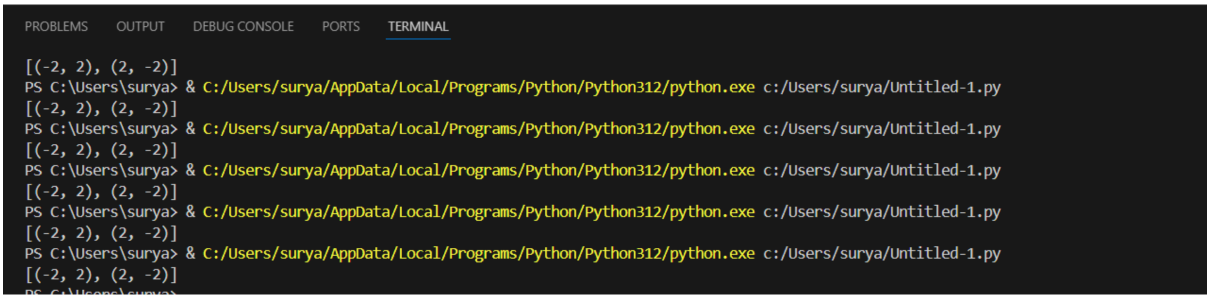
9. Given an array of points where points[i] = [xi, yi] represents a point on the X-Y plane and an integer k, return the k closest points to the origin (0, 0).The distance between two points on the X-Y plane is the Euclidean distance (i.e., √(x1 - x2)2 + (y1 - y2)2). You may return the answer in any order. The answer is guaranteed to be unique (except for the order that it is in).

```
import heapq

def kClosest(points, k):

  # Create a min-heap

  heap = []

  for (x, y) in points:

    distance = x**2 + y**2

    heapq.heappush(heap, (distance, (x, y)))

  result = []

  for _ in range(k):

    result.append(heapq.heappop(heap)[1])

  return result

points = [[1, 3], [-2, 2], [2, -2]]

k = 2

print(kClosest(points, k))
```

INPUT:[[1,3],[-2,2],[2,-2]]

OUTPUT:



```
[(-2, 2), (2, -2)]
PS C:\Users\surya> & C:/Users/surya/AppData/Local/Programs/Python/Python312/python.exe c:/Users/surya/Untitled-1.py
[(-2, 2), (2, -2)]
PS C:\Users\surya> & C:/Users/surya/AppData/Local/Programs/Python/Python312/python.exe c:/Users/surya/Untitled-1.py
[(-2, 2), (2, -2)]
PS C:\Users\surya> & C:/Users/surya/AppData/Local/Programs/Python/Python312/python.exe c:/Users/surya/Untitled-1.py
[(-2, 2), (2, -2)]
PS C:\Users\surya> & C:/Users/surya/AppData/Local/Programs/Python/Python312/python.exe c:/Users/surya/Untitled-1.py
[(-2, 2), (2, -2)]
PS C:\Users\surya> & C:/Users/surya/AppData/Local/Programs/Python/Python312/python.exe c:/Users/surya/Untitled-1.py
[(-2, 2), (2, -2)]
PS C:\Users\surya>
```

TIME COMPLEXITY:

O(nlogn)