K Closest Points to Origin

Try to solve the K Closest Points to Origin problem.

We'll cover the following

Statement

Given a list of points on a plane, where the plane is a 2-D array with (x, y) coordinates, find the k closest points to the origin $(0, 0)$.

> **Note:** Here, the distance between two points on a plane is the Euclidean distance: $\sqrt{x^2 + y^2}$

Constraints:

- $1 \leq$ `k` $\leq$ `points.length` $\leq 10^4$
- $-10^4 <$ `x[i]`, `y[i]` $< 10^4$

Examples

Sample example 1

Input

```
k = 1
```

```
(1, 3)  (-2, 2)
```

Output

```
(-2, 2)
```

1 of 4

Understand the problem

Let's take a moment to make sure you've correctly understood the problem. The quiz below helps you check if you're solving the correct problem:

K Closest Points to Origin

1 What is the output if the following list of points and value of k is given as input?

Select all that apply.

points = [[1, 3], [-2, 4], [2, -1], [-2, 2], [5, -3], [3, -2]]

k = 3

- [ ] **A)** [[1, 3], [-2, 2], [5, -3]]
- [ ] **B)** [[2, -1], [-2, 2], [1, 3]]
- [ ] **C)** [[1, 3], [-2, 2], [2, -1]]
- [ ] **D)** [[1, 3], [-2, 2], [3, -2]]

Submit Answer

Question 1 of 4
0 attempted

Reset Quiz

Figure it out!

We have a game for you to play. Rearrange the logical building blocks to develop a clearer understanding of how to solve this problem.

Drag and drop the cards to rearrange them in the correct sequence.

Push the first k points to the heap.

Calculate the distance between the origin and each point.

Compare the distance of the point with the distance of the top of the heap.

Push and pop the point from the heap.

Return the points from the heap.

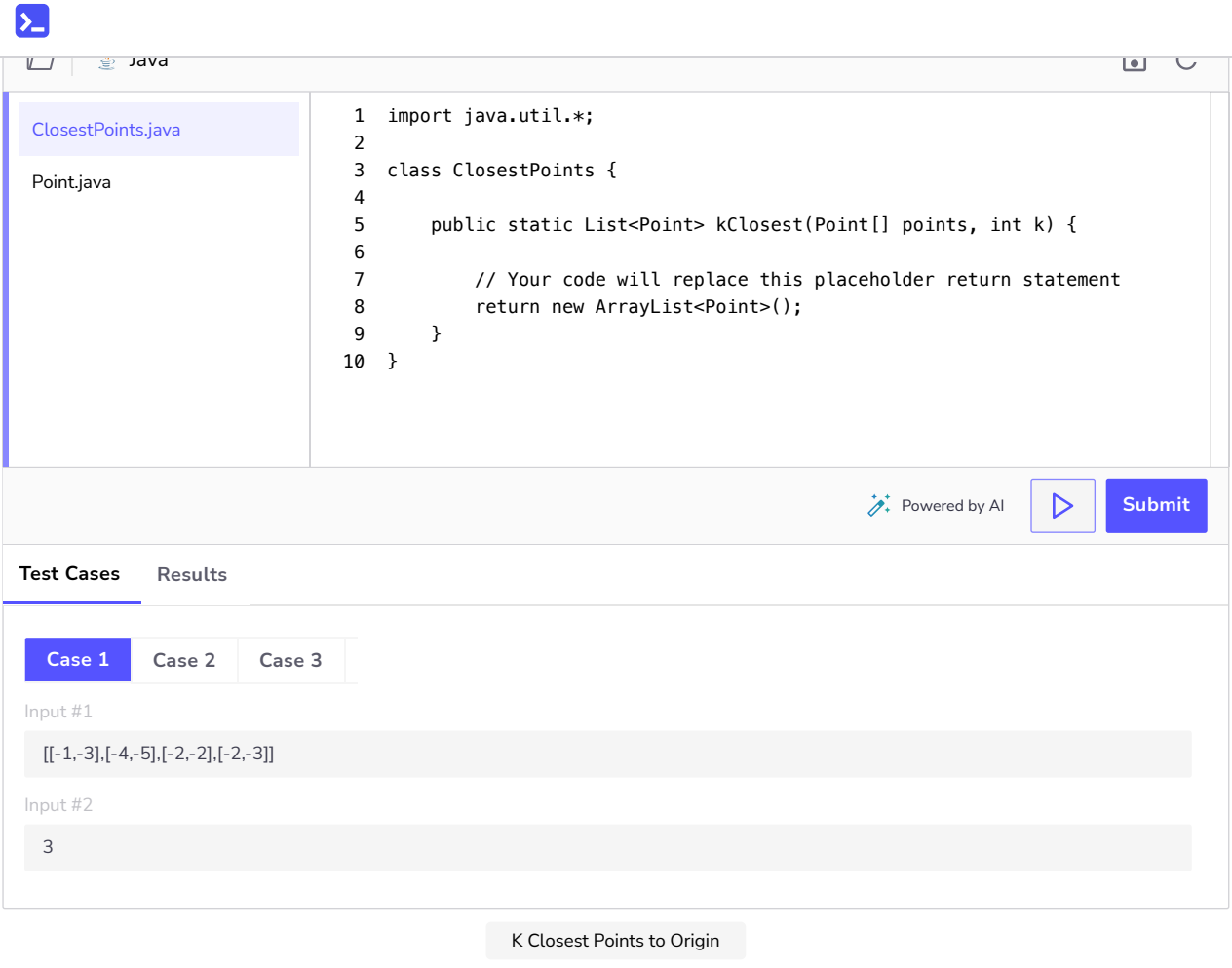
Reset | Show Solution | Submit

Try it yourself

Implement your solution in `ClosestPoints.java` in the following coding playground.

A `Point` class has two data members, `x` and `y` coordinates. You may add members or methods to it to support your solution.

> **Note:** The order of the points returned is not significant.



K Closest Points to Origin

Back
Solution: Reorganize S...

Next
Solution: K Closest Poi...

Mark as Completed