EXERCISE-85.Closest pair of points using divide and conquer.

Code:

```
import math
def dist(p1, p2):
    return math.sqrt((p1[0] - p2[0])**2 + (p1[1] - p2[1])**2)
def brute_force(points, n):
    min_dist = float('inf')
    for i in range(n):
        for j in range(i + 1, n):
            if dist(points[i], points[j]) < min_dist:
                min_dist = dist(points[i], points[j])
    return min_dist
def strip_closest(strip, size, d):
    min_dist = d
    strip.sort(key=lambda point: point[1])
  for i in range(size):
        for j in range(i + 1, size):
            if (strip[j][1] - strip[i][1]) < min_dist:
                min_dist = dist(strip[i], strip[j])
    return min_dist
def closest_util(points, n):
    if n <= 3:
        return brute_force(points, n)
    mid = n // 2
  mid_point = points[mid]
    dl = closest_util(points[:mid], mid)
    dr = closest_util(points[mid:], n - mid)
    d = min(dl, dr)
    strip = []
    for i in range(n):
        if abs(points[i][0] - mid_point[0]) < d:
```

```
            strip.append(points[i])
    return min(d, strip_closest(strip, len(strip), d))
def closest(points):
    points.sort(key=lambda point: point[0])
    return closest_util(points, len(points))
points = [(2, 3), (12, 30), (40, 50), (5, 1), (12, 10), (3, 4)]
print("The smallest distance is", closest(points))
```

Output:

```
The smallest distance is 1.4142135623730951
```