**Approach: BRUTE FORCE**

1) To check distance between all four sides must be equal

2) For a rhombus all 4 sides are equal but adjacent sides are not perpendicular( not 90 degree)

3) Therefore a square must follow both conditions:

4) To consider all possibilites of arrangements: 4points can be arranged amongst themselves in 4!=24ways

5) To check every possible permutation

Time-> O(1)

Space -> O(1)

**Approach: Check every case**

1)Instead of going to all permutations we can see that few set of permutations give same result so 24 arrangements are reduced to 3 sets of arrangements.

eg : P1P2P3P4 => P2P3P4P1

so check only 3 cases:

a) p1p2p3p4 -> (p1,p2) (p2,p3) (p3,p4) (p4,p1) and (p1,p3) == (p2,p4) diagonals equal

b) p1p3p2p4 -> (p1,p3) (p3,p2) (p2,p4) (p4,p1)

c) p1p2p4p3

Code:

def find\_dist(pa,pb):

dx = pow(pa[0]-pb[0], 2)

dy = pow(pa[1]-pb[1], 2)

dist = dy+dx

return dist

def check(a,b,c,d):

return dist(a,b) > 0 and dist(a,b) == dist(b,c)==dist(c,d) == dist(d,a) and dist(a,c) == dist(b,d) and a!=c and b!=d

return check(p1,p2,p3,p4) or check(p1,p2,p4,p3) or check(p1,p3,p2,p4)