Why my programs do not deadlock (for reasonable choices of input parameters)?

For Q1A:

Basically, our program may encounter deadlock because when acquire block on two points to add an edge. When a thread gets a point's lock, another thread probably had the other point's lock, and both of the threads try to get each other's point's lock, encountering deadlock.

In class, we learned dining philosopher problem, and the optimal solution is ordering the resources. I also use this technic in my program. I indexed all points I created and the default four points. Then I ask all threads must obey obtain smaller point's lock first. This way, even two threads wants add edges on same pair of points and acquire both locks, they start on same point, thus there is always a first thread to acquire lock, making another thread can only wait.

For Q1B:

Same as q1a, just we need order four points at once instead of two. Thus, every spider object in my program will attempt four points together, and if there are two spiders trying to get four same points, they will start from smallest one. So, won't happen deadlock.