Finding collinear points

The problem: Given a set of n points in the plane, determine if there exist three points that are collinear.

We sketched the following ideas:

Algorithm 1 (brute force)

- for all distinct triplets of points p_i, p_j, p_k: if collinear return true
- (if you get here) return false

Algorithm 2

- initialize array L = empty
- for all distinct pairs of points p_i, p_j
  - compute their line equation (slope, intercept) and add it to an array L
- sort array L by (slope, intercept)
- traverse L and if you find any 3 consecutive identical (s,i) $\rightarrow$ collinear

Algorithm 3

- initialize BBST = empty
- for all distinct pairs of points p_i, p_j
  - compute their line equation (s, i)
  - insert (s,i) in BBST; if when inserting you find that (s,i) is already in the tree, you got three collinear points and return true
- (if you ever get here) return false

Algorithm 4

- initialize HashTable = empty
- for all distinct pairs of points p_i, p_j
  - compute their line equation (s, i)
  - insert (s,i) in HashTable; if when inserting you find that (s,i) is already in the HT, you got three collinear points and return true
- (if you ever get here) return false

Algorithm 5

- for every point p_i
  - set array L = empty
  - for every point p_j (with $p_j != p_i$)
    - compute slope of p_j wrt to p_i and add it to array L
  - sort L
  - traverse L and if you find two consecutive points that have same slope, they are collinear with p_i so return true
- (if you get here) return false

Questions:

1. What is the running time of Algorithm 1?
2. Argue briefly that Algorithm 2 is correct.
3. What is the running time of Algorithm 2 and how much space does it use?
4. What is the running time of Algorithm 3 and how much space does it use?
5. How does its worst/best case compare to Algorithm 2?
6. What is the running time of Algorithm 4 and how much space does it use?
7. Hoes does its worst/best case compare to Algorithm 3?
8. Under what assumption on the input is Algorithm 4 faster than Algorithm 3?
9. Argue briefly that Algorithm 5 is correct.
10. What is the running time of Algorithm 5 and how much space does it use?