Pot Burst Problem

Problem: N number of pots are kept at different heights. You are asked to find out number of arrows to burst them. When an arrow hits the pot it goes one level down. Assume that the pots are having same size for example given the pots heights as array (Array will be given in decreasing order of size) :5 4 3 3 2 2 1 1 1 minimum number of arrows to shoot them is: 3

Algorithm:

1. Start
2. **if** (potPositionArray.length == 0) then return 0.
3. **Sort array in descending order**
4. Initialize sameNum = 0, seq = 1, prev = 0
5. **Check for** (**int** i = 1; i < potPositionArray.length; i++) do
6. **if** (potPositionArray[i - 1] - potPositionArray[i] > 1) then seq++
7. **if** (potPositionArray[i] == potPositionArray[i - 1]) then sameNum++
8. **if** (potPositionArray[i] != potPositionArray[i - 1]) then sameNum = 0
9. **if** (prev < sameNum) then prev = sameNum
10. **end for**
11. return seq + prev
12. Stop

Pictures:

Proper input: 5 4 3 3 2 2 1 1 1

Output : 3 arrows required

4

5

3

3

2

2

1

1

1