

Highway Billboard

$m = 20$ (highway length)

$n = 5$ (count of bill-boards)

6 7 12 14 18 (posⁿ of bill-board)

5 8 5 3 1 (revenue by billboard)

3 (gap)

$m = 20$ (highway length)

$n = 5$ (count of bill-boards)

6 7 12 14 18 (posⁿ of bill-board)

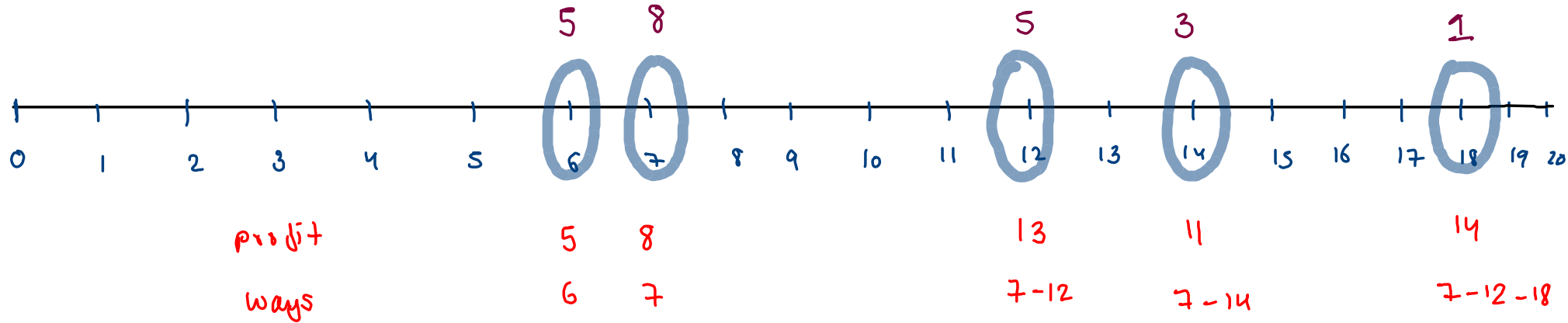
5 8 5 3 1 (revenue by bill-board)

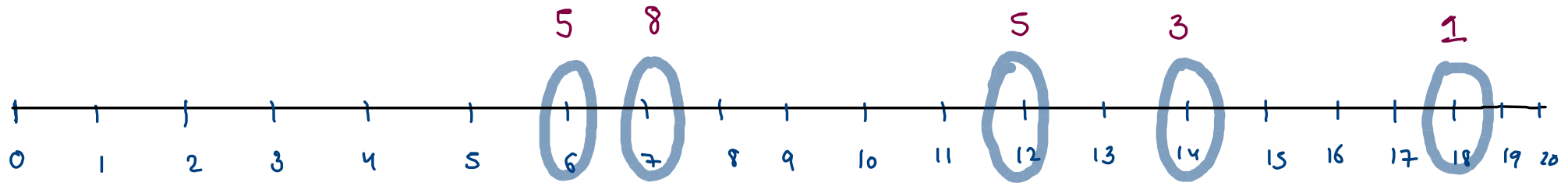
3 (gap)

(i) LIS (bill boards posⁿ)

$T : n^2$

$n^2 < m$





```
int[] dp = new int[x.length];
dp[0] = rev[0];
int omax = 0;

for(int i=1; i < dp.length; i++) {
    int max = 0;
    for(int j=0; j < i; j++) {
        int dist = x[i] - x[j];

        if(dist > t) {
            max = Math.max(dp[j], max);
        }

        dp[i] = max + rev[i];
        omax = Math.max(omax, dp[i]);
    }
}
```

$t = 5$

n^2

x is sorted

x

6	7	12	14	18
0	1	2	3	4

dp

5	8	10	11	11
0	1	2	3	4

ways 6 - 12 - 18

7 - 14

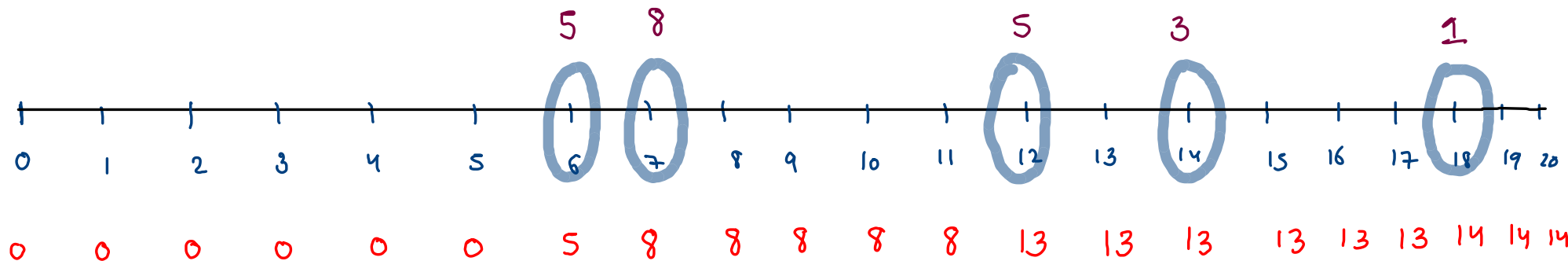
x is sorted then work directly otherwise
sort it.

(ii) miles

$$m < n^2$$

Time $O(m)$

(include & exclude)



$dp[i-1]$

$row + dp[i-gap-1]$

exc

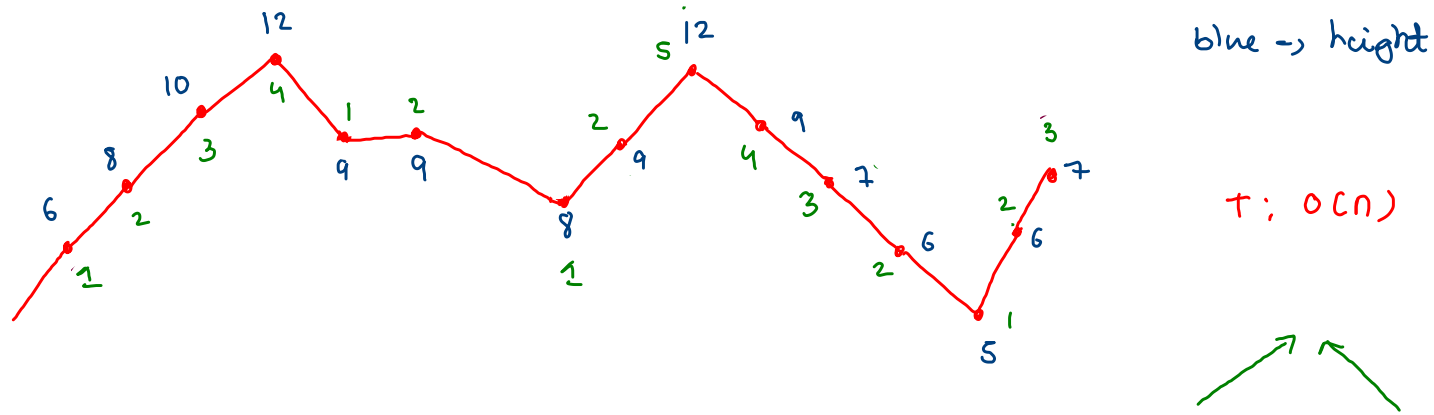
inc

$dp[i]$

gap = 3

Temple Offerings

1. Pepcoder is wishing to give offerings to all the temples along a mountain range.
2. The temples are located in a row at different heights.
3. You have to find the minimum number of offerings such that these conditions are fulfilled -
 - > If two adjacent temples are at different heights, then the temple which is situated at greater height should receive more offerings.
 - > If two adjacent temples are at the same height, then their offerings relative to each other does not matter.



	6	8	10	12	9	9	8	9	12	9	7	6	5	6	7
left	1	2	3	4	1	1	1	2	3	1	1	1	2	2	3
right	1	1	1	2	1	2	1	1	5	4	3	2	1	1	1
off	1	2	3	4	1	2	1	2	5	4	3	2	1	2	3

max (left[i], right[i]) → sum