

2.

HighestProfit(A, i, j)

$$\{ \text{if } (i=j) \text{ return } (A[i], A[j], 0)$$

else

$$\{ k \leftarrow \frac{(i+j)}{2}$$

$$(low-L, high-L, profit-L) \leftarrow \text{HighestProfit}(A, \underline{i}, \underline{k})$$

$$(low-R, high-R, profit-R) \leftarrow \text{HighestProfit}(A, \underline{k+1}, \underline{j})$$

$$low \leftarrow \underline{\min(low-L, low-R)}$$

$$high \leftarrow \underline{\max(high-L, high-R)}$$

$$profit \leftarrow \underline{\max(profit-L, profit-R, high-R - low-L)}$$

$$\text{return } (low, high, profit);$$

}

The time complexity of algorithm is $O(n)$
 (where n is the no. of days.)

Final Answers

a) i, k b) $k+1, j$ c) $\min(low-L, low-R)$ d) $\max(high-L, high-R)$ e) $\max(profit-L, profit-R, high-R - low-L)$ f) $O(n)$

Proof:-

$$\checkmark T(n) = 2T\left(\frac{n}{2}\right) + C$$