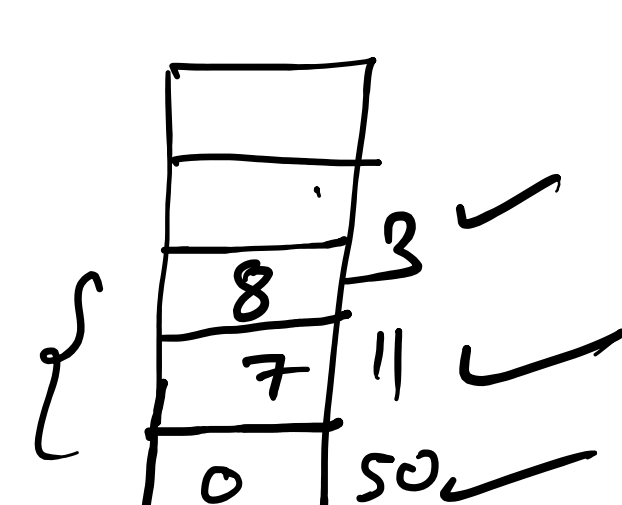


①

$$\begin{array}{cccccccccccc} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ [5, 0, 5, 9, 2, 8, 7, 9, 11, 3] \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ -1 & 8 & 8 & 8 & 9 & 9 & 11 & -1 & -1 \end{array}$$

$$\begin{array}{cccccccccccc} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ [5, 0, 5, 9, 2, 8, 7, 9, 11, 3] \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ -1 & 8 & 8 & 8 & 9 & 9 & 11 & -1 & -1 \end{array}$$

- next sister ✓
- next smaller
- prev ~~smaller~~ sister
- prev smaller



arr[i] > arr[st+fee]

8 > 2

5 > 50

3 > 5

3 > 2

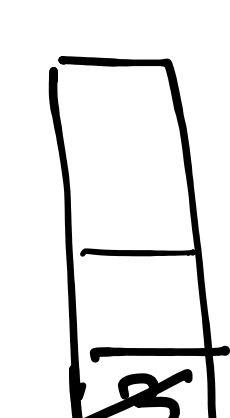
8 > 2 =

```

public static void NGE(int[] arr) {
    Stack<Integer> st = new Stack<>();
    int[] ans = new int[arr.length];
    for (int i = 0; i < arr.length; i++) {
        while (!st.isEmpty() && arr[st.peek()] < arr[i]) {
            ans[st.pop()] = arr[i];
        }
        st.push(i);
    }
    while (!st.isEmpty()) {
        ans[st.pop()] = -1;
    }
    // nge
    for (int i = 0; i < ans.length; i++) {
        System.out.println(arr[i] + " " + ans[i]);
    }
}

```

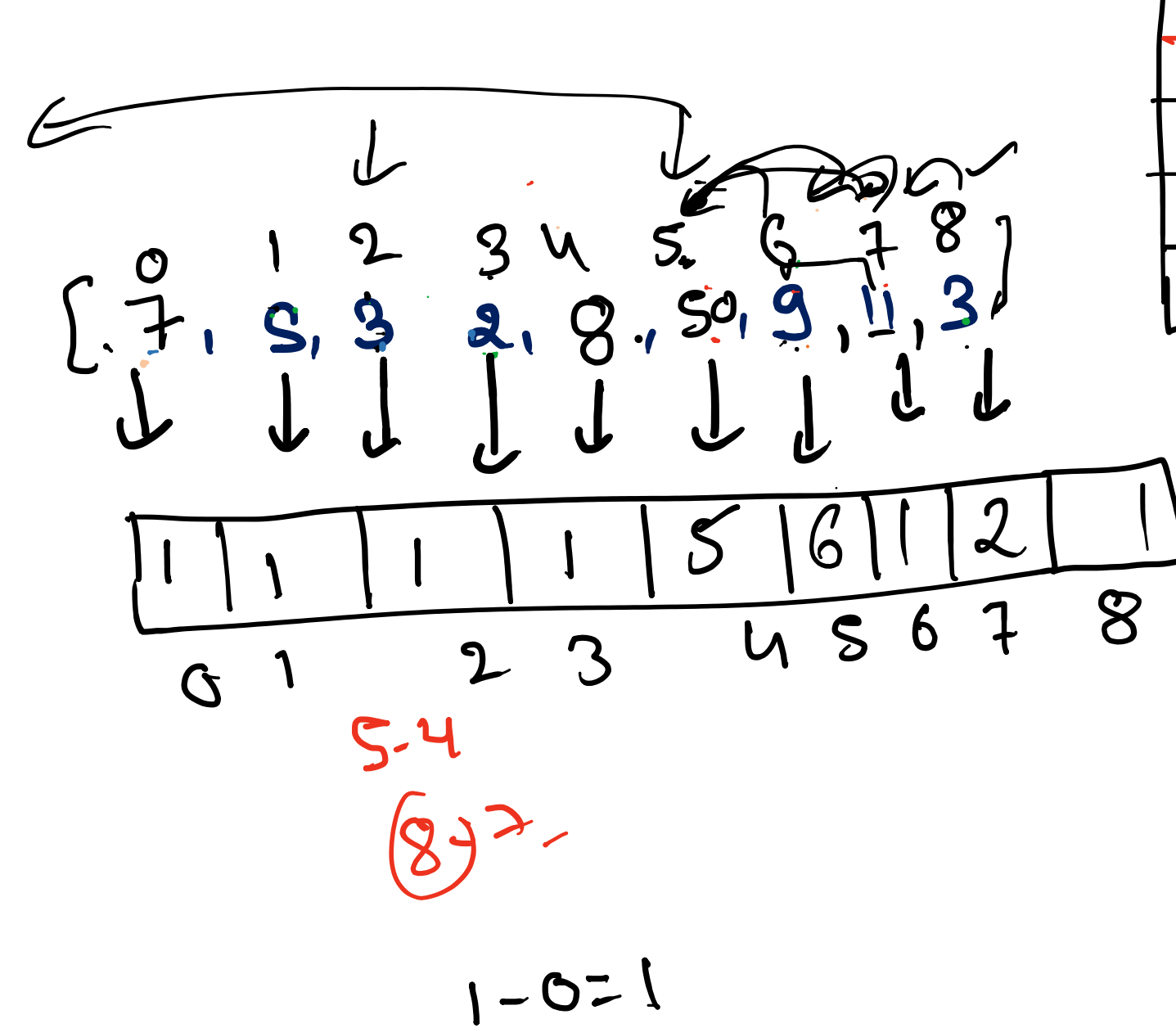
$[4, 5, 2, 10]$
 $[5 | 10 | 10 | -]$



The stock span problem is a financial problem where we have a series of N daily price quotes for a stock and we need to calculate span of stock's price for all N days. You are given an array of length N , where i^{th} element of array denotes the price of a stock on i^{th} . Find the span of stock's price on i^{th} day, for every $1 \leq i \leq N$.

A span of a stock's price on a given day, i , is the maximum number of consecutive days before the $(i+1)^{\text{th}}$ day, for which stock's price on these days is less than or equal to that on the i^{th} day.

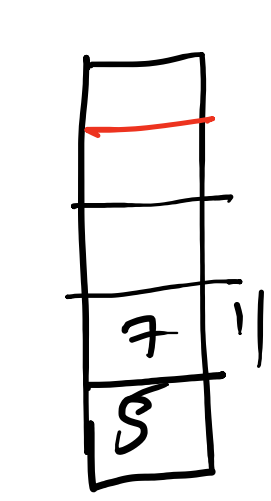
Handwritten diagram of a binary tree for the sequence 30, 35, 40, 38, 35. The root is 40, with children 30 and 38. 30 has children 0 and 1, and 38 has children 2 and 3. Node 0 has child 1, and node 3 has child 4. Red arrows indicate a path from 40 to 0 to 1, and from 40 to 2 to 3 to 4. A red arrow also points from the root 40 to the left child 30.

$$[arr[i]] > \underbrace{\text{left side}}_{50 \leq 5}$$


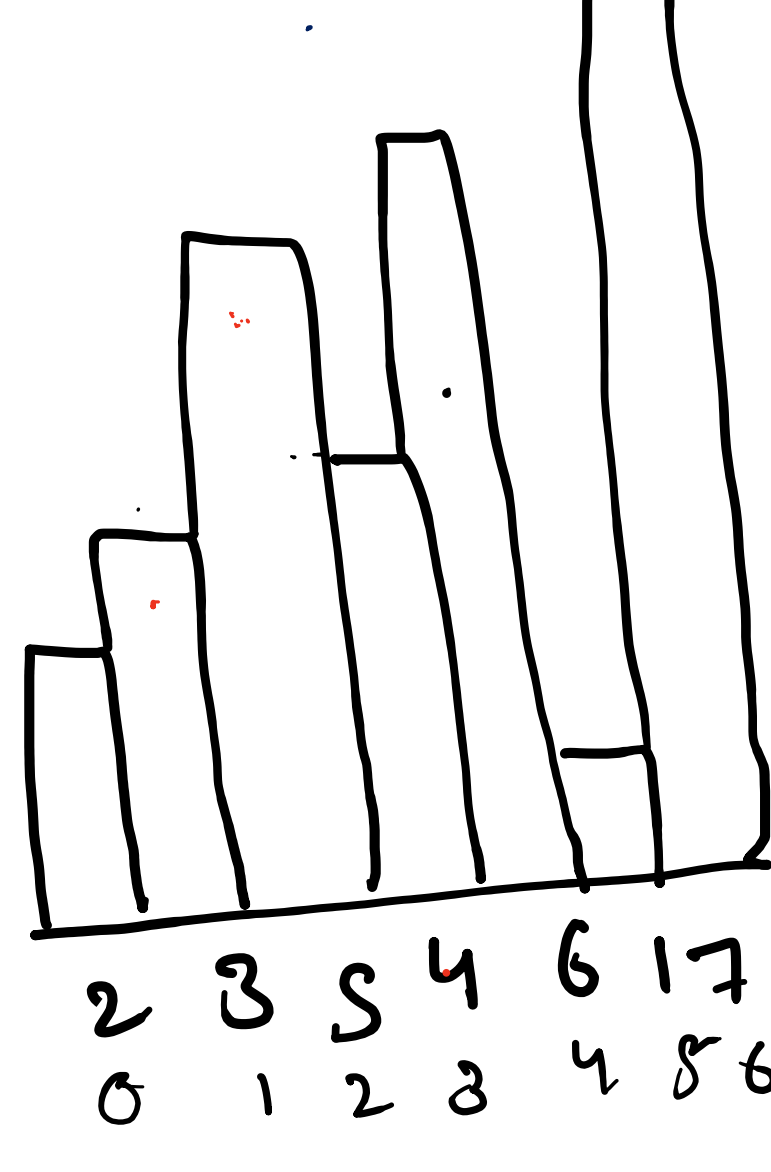
```

    3
    // space cal
    if (st.isEmpty()) {
        ans[i] = i + 1
    } else {
        ans[i] = i - st.peek()
    }
    st.push(i)

```



0 1 2 3 4 5 6


$$\left. \begin{array}{l} 2 \times 5 = 10 \\ 3 \times 4 = 12 \\ 5 \times 1 = 5 \\ 4 \times 3 = 12 \\ 6 \times 1 = 6 \\ 1 \times 7 = 7 \\ 7 \times 1 = 7 \end{array} \right\} (12)$$