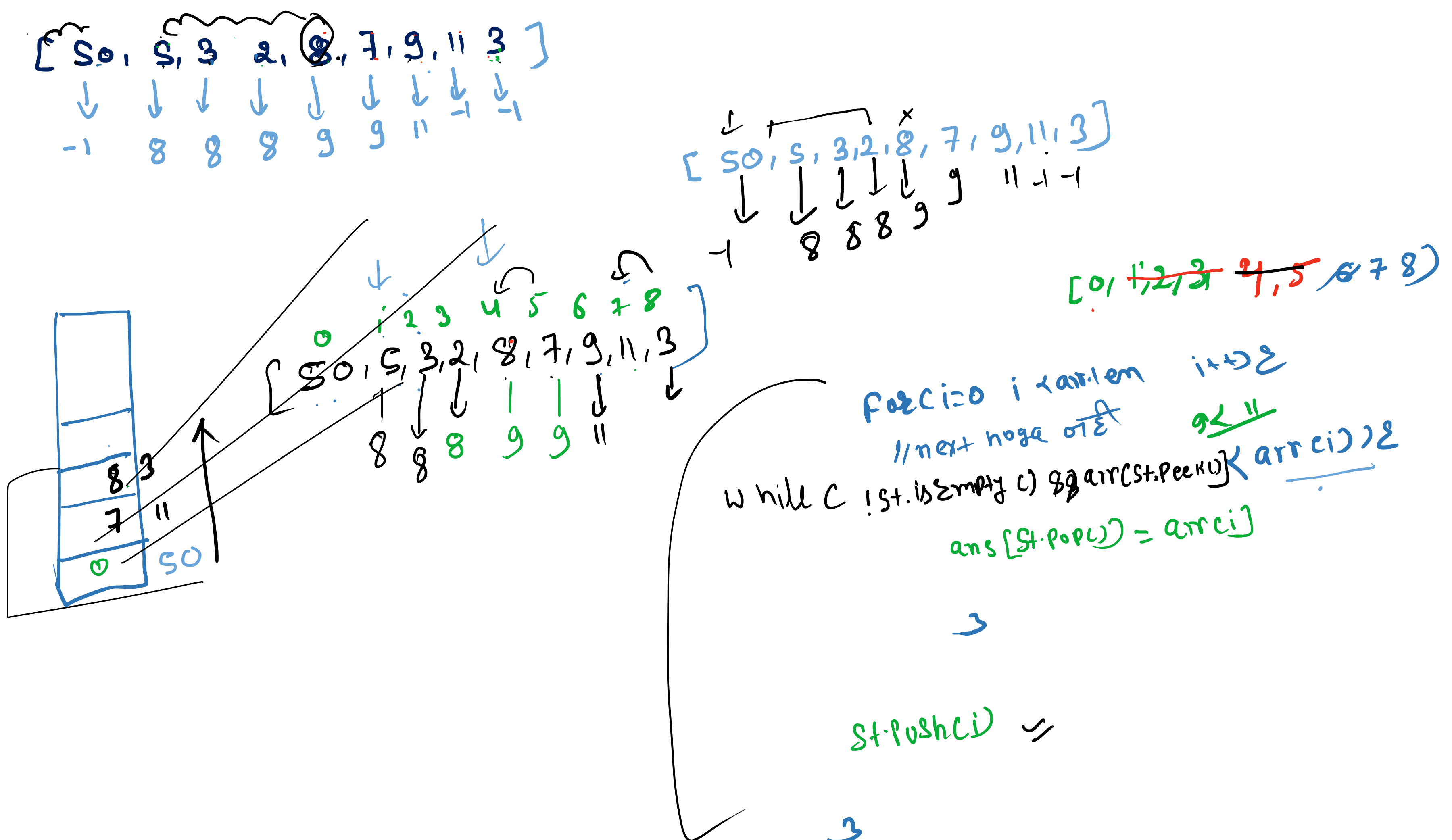
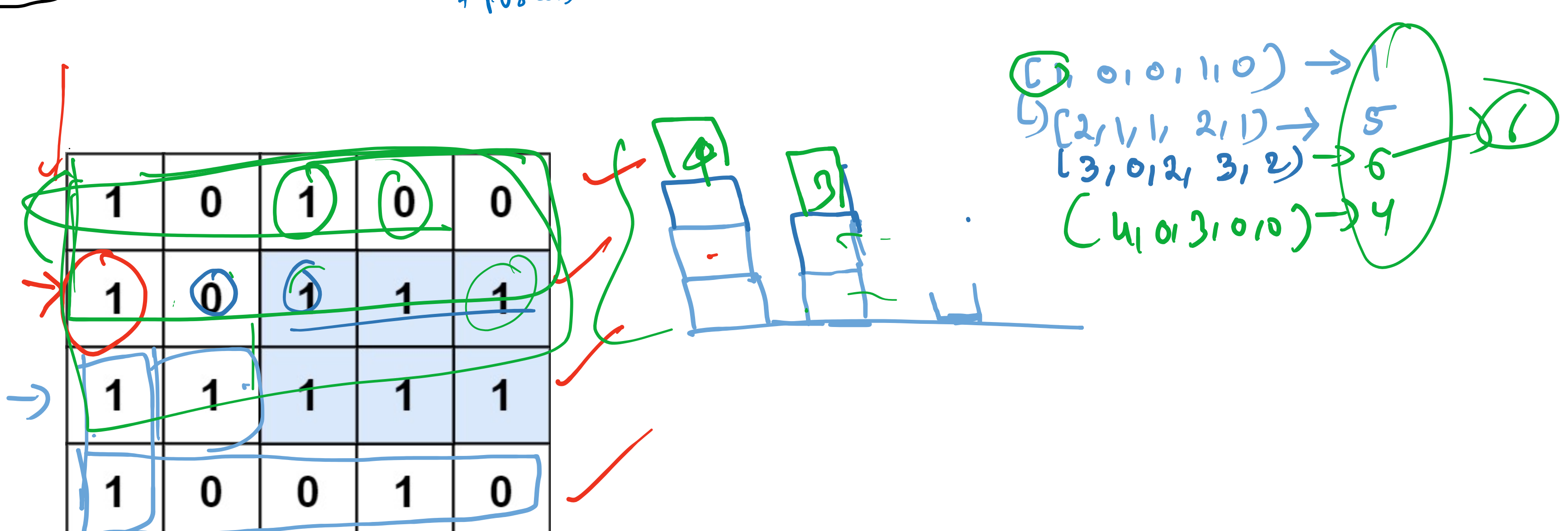
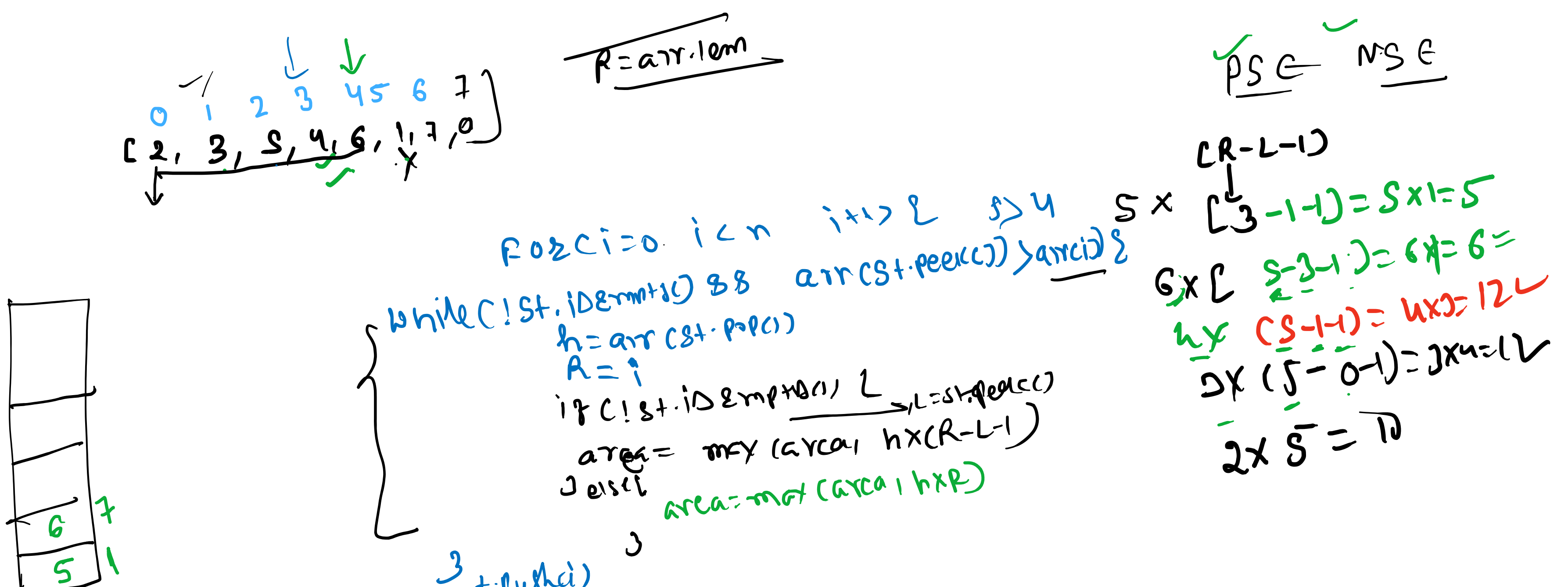
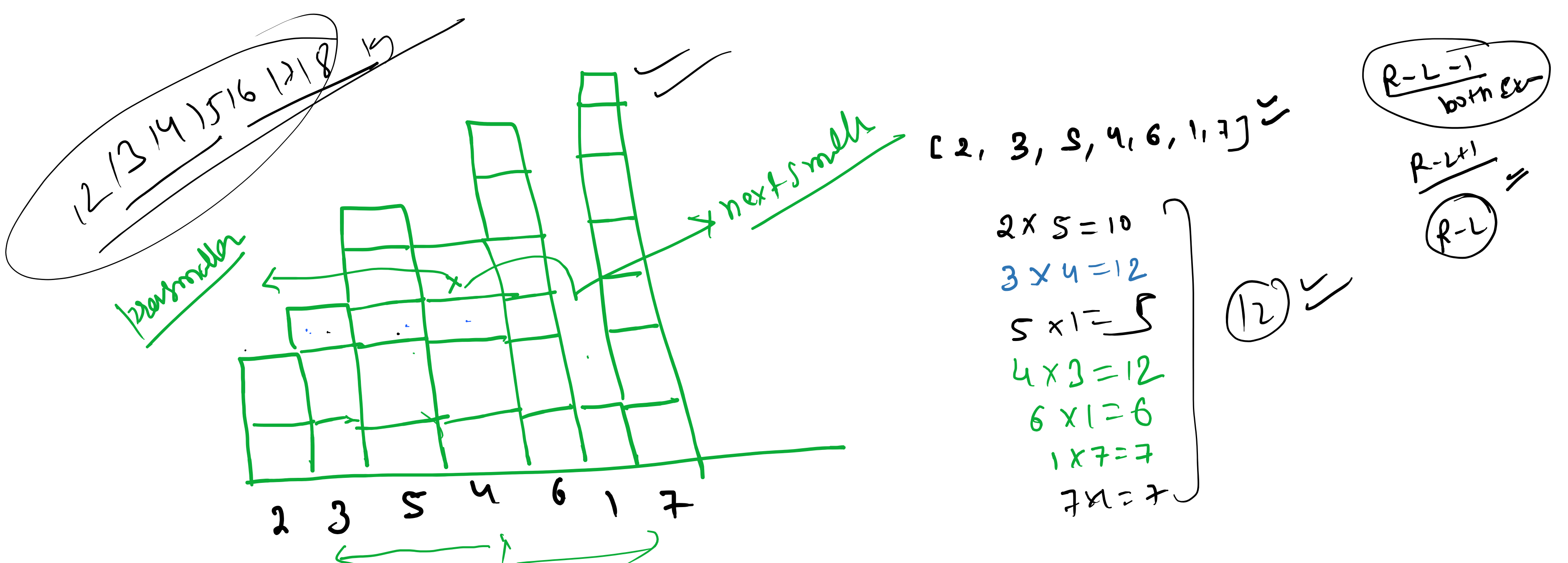
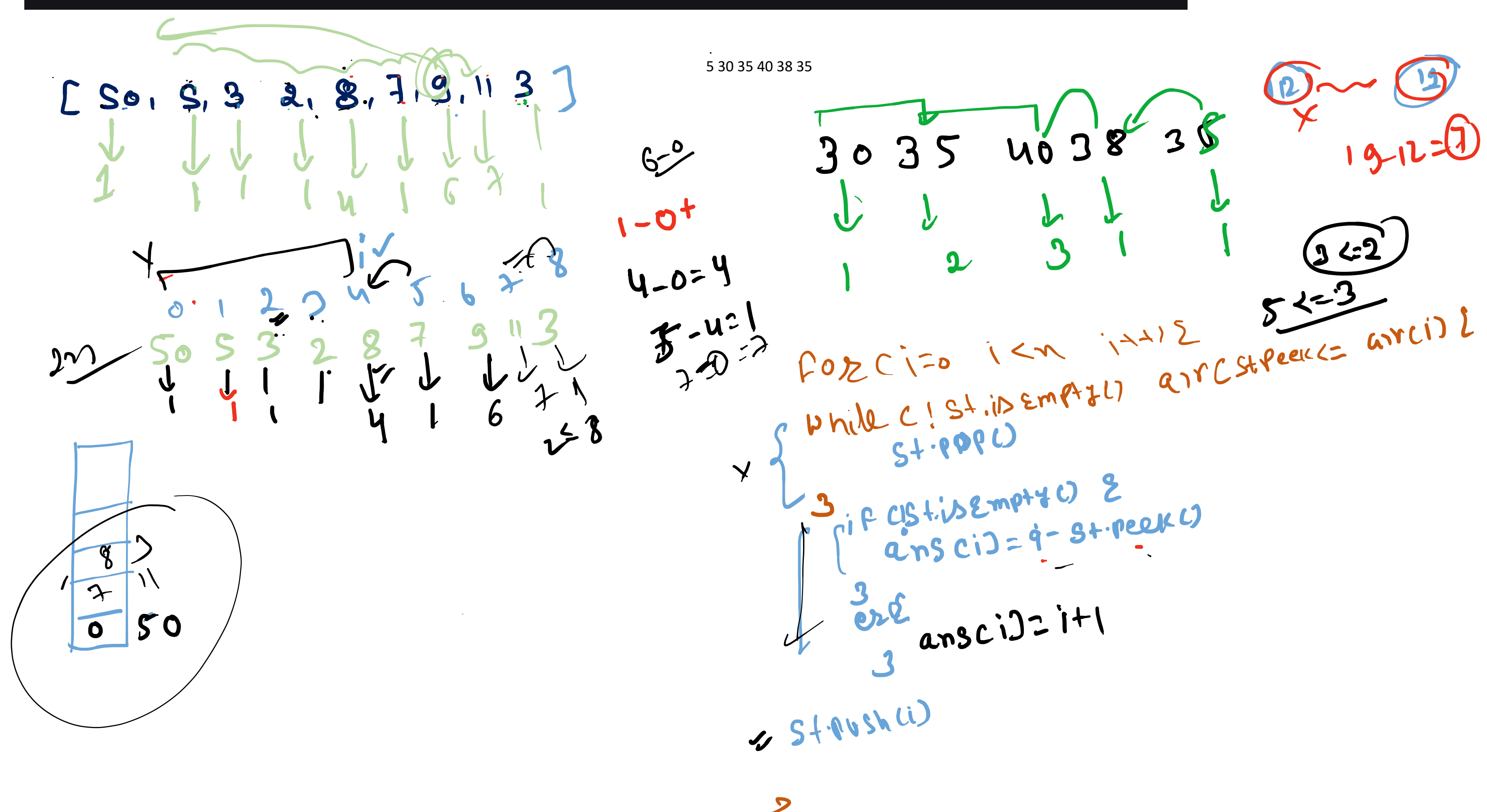


OC(n)

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 $1 \leq i \leq N$.

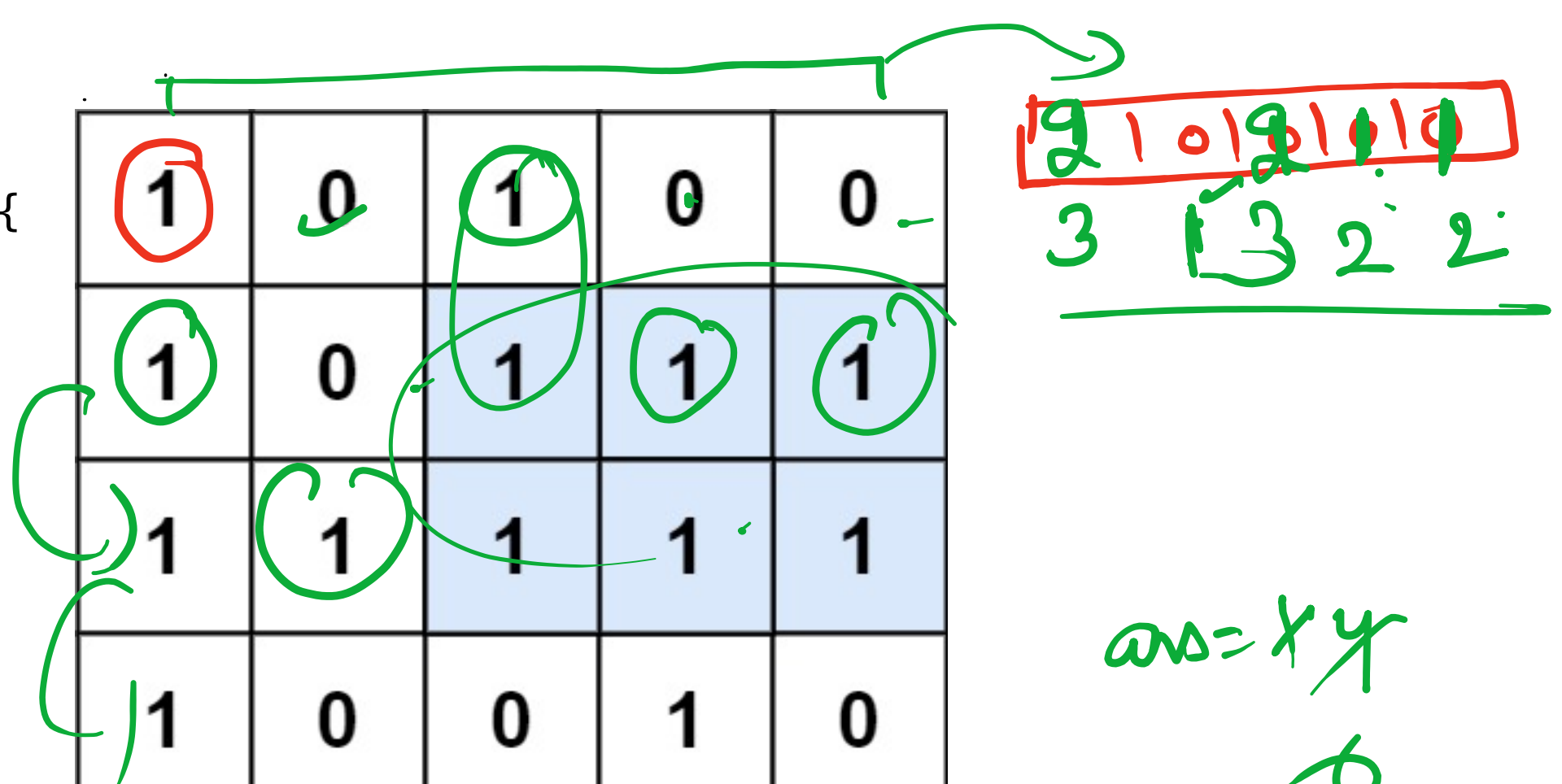
A span of a stock's price on a given day, i , is the maximum number of consecutive days before the $(i+1)^{\text{th}}$ day, for which stock's price on these days is less than or equal to that on the i^{th} day.



```

int [] arr = new int [matrix[0].length];
int ans=0;
for (int i = 0; i < matrix.length; i++) {
    for (int j = 0; j < matrix[0].length; j++) {
        if (matrix[i][j] == '1') {
            arr[j]++;
        } else {
            arr[j] = 0;
        }
    }
    ans = Math.max(ans, Maximum_Area(arr));
}

```



Given a string s containing just the characters '(', ')', '{', '}', '[', and ']', determine if the input string is valid.

An input string is valid if:

- Open brackets must be closed by the same type of brackets.
- Open brackets must be closed in the correct order.
- Every close bracket has a corresponding open bracket of the same type.

3

Handwritten notes for the Valid Parentheses problem:

String: $[(23)]$

Diagram showing a stack and the calculation of span for each element.

