

## Intuition

Let's take a look on this interesting problem.

For our candies task we have **two crucial requirements**: Each child must have **at least one** candy, and children with **higher ratings** than their **neighbors** have more candies than them... Interesting.

Let's explore some ratings' cases together:

2, 2, 2

Here we can see that **each child** of them can have **one candy** since no one has higher rating than its neighbor.

1, 2, 2, 2

Here we can see that the array of candies will be 1, 2, 1, 1.

9, 8, 7

Here we can see that array of candies will be 3, 2, 1.

7, 8, 9

Here we can see that array of candies will be 1, 2, 3.

I think now we have some **idea** about what is going on here 🤔

- Children with **equal ratings** can have different candies and that won't violate our requirements since we are aiming for giving **minimum** candies.
- We must take care of **decreasing** subarrays and **increasing** ones.

How can we use an approach that take care of these **observations**? 🚀

### Two Pass Approach

In this approach we will:

- Iterate from left to right to take care of **increasing** subarrays
- Iterate from right to left to take care of **decreasing** subarrays
- We **won't do anything** to **equal** neighbours since they won't affect our requirements

Let's take an example:

9, 8, 8, 7, 7, 8, 9

**First** we have array of candies 1, 1, 1, 1, 1, 1, 1

After our **first pass** (from left to right) we will have array of candies 1, 1, 1, 1, 1, 2, 3.

We can see that it **handled** the **increasing** subarray perfectly.

After our **second pass**(from right to left) we will have array of candies 2, 1, 2, 1, 1, 2, 3 .

we can see that it also **handled** the **decreasing** subarray perfectly.

Also, it handled the equal neighbours after the two passes. It finally minimized our candies.

## **Complexity**

- **Time complexity:**  $O(N)O(N)O(N)$

Since we are looping over the ratings array twice then the time complexity is  $2*N$  which is  $O(N)$ .

- **Space complexity:**  $O(N)O(N)O(N)$

Since we are storing the number of candies for each child then the space complexity is  $O(N)$ .