455 Assign Cookies (link)

Description

Assume you are an awesome parent and want to give your children some cookies. But, you should give each child at most one cookie.

Each child `i` has a greed factor `g[i]`, which is the minimum size of a cookie that the child will be content with; and each cookie `j` has a size `s[j]`. If `s[j] >= g[i]`, we can assign the cookie `j` to the child `i`, and the child `i` will be content. Your goal is to maximize the number of your content children and output the maximum number.

Example 1:

```
Input: g = [1,2,3], s = [1,1]
Output: 1
Explanation: You have 3 children and 2 cookies. The greed factors of 3 children are 1
And even though you have 2 cookies, since their size is both 1, you could only make tl
You need to output 1.
```

Example 2:

```
Input: g = [1,2], s = [1,2,3]
Output: 2
Explanation: You have 2 children and 3 cookies. The greed factors of 2 children are 1
You have 3 cookies and their sizes are big enough to gratify all of the children,
You need to output 2.
```

Constraints:

- `1 <= g.length <= 3 *` 10^4
- `0 <= s.length <= 3 *` 10^4
- `1 <= g[i], s[j] <=` 2^{31} `- 1`

(scroll down for solution)

Solution

Language: cpp

Status: Accepted

```
#include <vector>
#include <algorithm>

using namespace std;

class Solution {
public:
    int findContentChildren(vector<int>& g, vector<int>& s) {
        sort(g.begin(), g.end()); // Сортируем жадность детей
        sort(s.begin(), s.end()); // Сортируем размеры печенек

        int i = 0; // Индекс для жадности детей
        int j = 0; // Индекс для размеров печенек
        int satisfiedChildren = 0; // Счетчик удовлетворенных детей

        // Проходим по обоим массивам
        while (i < g.size() && j < s.size()) {
            if (s[j] >= g[i]) { // Если размер печеньки удовлетворяет жадности ребенка
                satisfiedChildren++;
                i++; // Переходим к следующему ребенку
            }
            j++; // В любом случае переходим к следующей печеньке
        }

        return satisfiedChildren;
    }
};
```