

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
ИТМО»

Отчёт по лабораторной работе № 6
«Динамическое программирование»

Выполнил работу

Фиолетов Э.А.

Академическая группа №3114

Принял

Дунаев М.В.

Санкт-Петербург

2024

Отчет по лабораторной работе

1. Асимптотика и память

Временная сложность:

1. Заполнение `left_max` требует одного прохода по массиву $O(n)$.
2. Заполнение `right_max` требует одного обратного прохода $O(n)$.
3. Один проход по массиву для подсчёта воды $O(n)$.

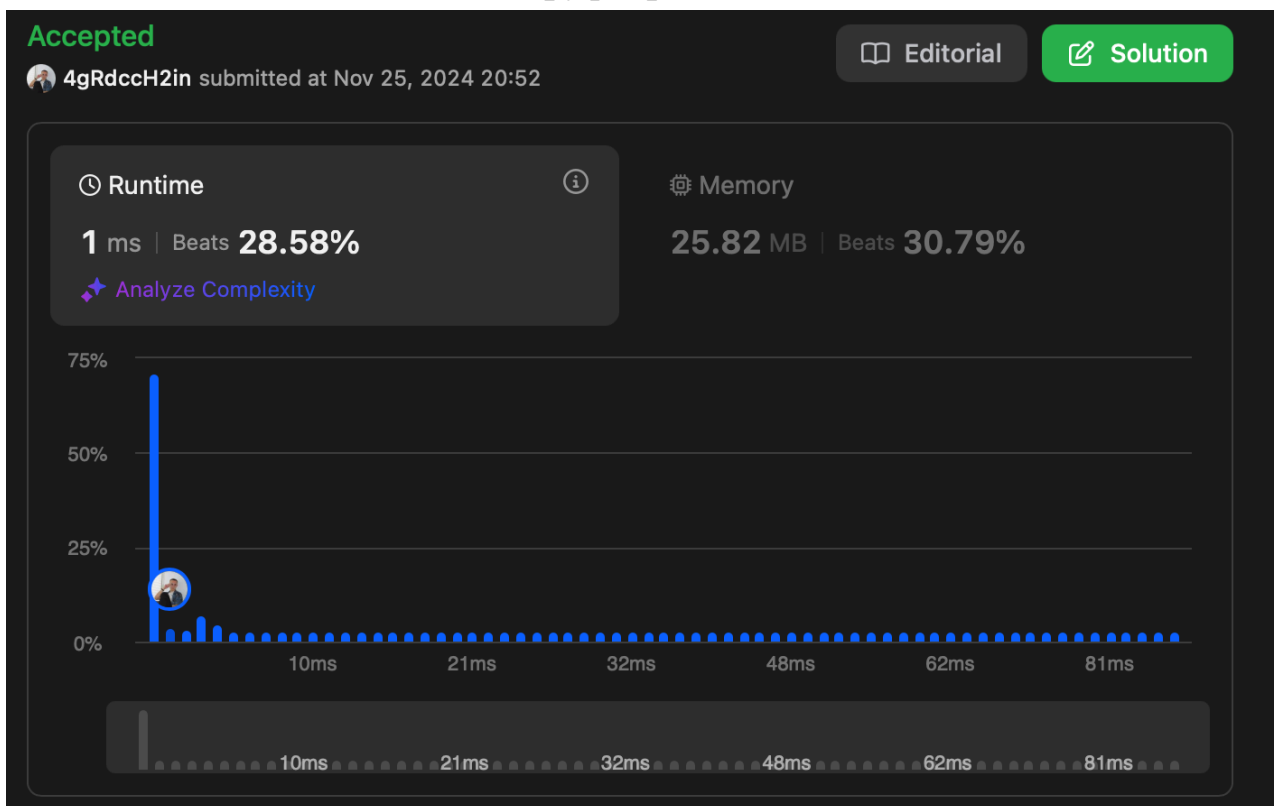
Итого: $O(n) + O(n) + O(n) = O(n)$.

Пространственная сложность:

1. Входные данные: Сам входной массив `height` занимает $O(n)$.
2. `left_max`: требует $O(n)$.
3. `right_max`: требует $O(n)$.

Итого: $O(n) + O(n) + O(n) = O(n)$.

2. Пруфы решения



3. Код решения

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

class Solution {
public:
    int trap(std::vector<int>& height) {
        int n = height.size();
        if (n == 0) return 0;

        std::vector<int> left_max(n);
        std::vector<int> right_max(n);

        left_max[0] = height[0];
        for (int i = 1; i < n; ++i) {
            left_max[i] = std::max(left_max[i - 1], height[i]);
        }

        right_max[n - 1] = height[n - 1];
        for (int i = n - 2; i >= 0; --i) {
            right_max[i] = std::max(right_max[i + 1], height[i]);
        }

        int water = 0;
        for (int i = 0; i < n; ++i) {
            water += std::min(left_max[i], right_max[i]) - height[i];
        }

        return water;
    }
};
```

4. Пояснения

Данную задачу нужно решать с помощью дп так как при других способах асимптотика возрастает до $O(n^2)$, сам код получается сложнее. В случае дп нужно лишь 3 раза обойти собранный массив, что все еще линейно по асимптотике. Я даже не знаю что еще про эту задачу написать так что держи смешнявку



**В 7 лет учу делать
нейросеть на Pytorch**



ПРОГРАММИРУЮ В 5 ЛЕТ
4.1K views • 2 weeks ago