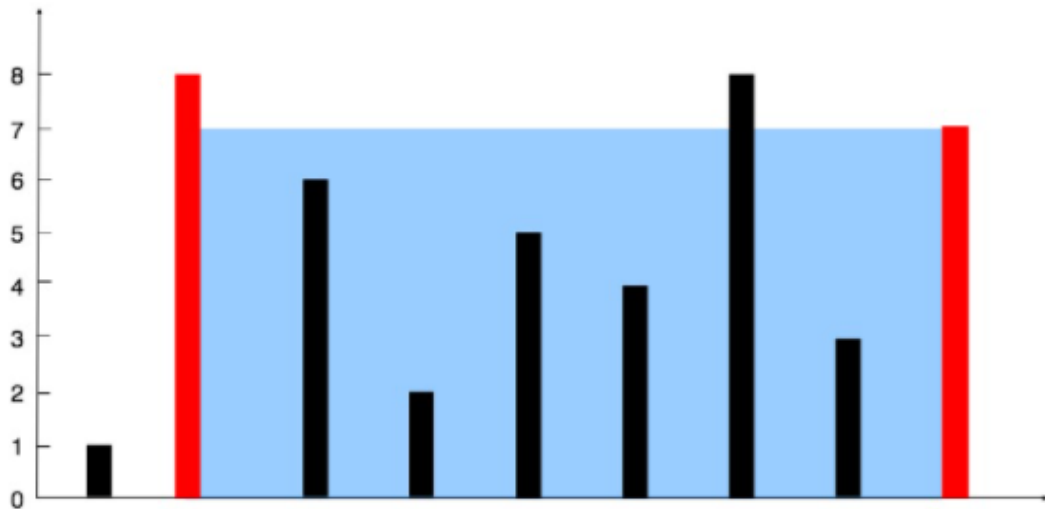


盛最多水的容器

给你 n 个非负整数 a_1, a_2, \dots, a_n ，每个数代表坐标中的一个点 (i, a_i) 。在坐标内画 n 条垂直线，垂直线 i 的两个端点分别为 (i, a_i) 和 $(i, 0)$ 。找出其中的两条线，使得它们与 x 轴共同构成的容器可以容纳最多的水。

说明：你不能倾斜容器。

示例 1：



输入：[1,8,6,2,5,4,8,3,7]

输出：49

解释：图中垂直线代表输入数组 [1,8,6,2,5,4,8,3,7]。在此情况下，容器能够容纳水（表示为蓝色部分）的最大值为 49。

示例 2：

输入：height = [1,1]

输出：1

示例 3：

输入：height = [4,3,2,1,4]

输出：16

示例 4：

输入：height = [1,2,1]

输出：2

```
class Solution {
public:
    int maxArea(vector<int>& height) {
        //双指针
        int i=0;
        int j=height.size()-1;
        int res=0;
        while(i<j)
        {
            if(height[i]<height[j])
            {
                res=max(res,height[i]*(j-i));
                i++;
            }
            else
            {
                res=max(res,height[j]*(j-i));
                j--;
            }
        }
        return res;
    }
};
```