解题思路：

这个题目采用两根指针来解决，因为要找最大的可以容纳水的面积，所以从两端向中间找，一定能够找到最大的解。接下来，这个问题就比较简单了，每次只需要计算首尾两端指针对应的最小值作为桶的高度，指针之间的距离作为桶的宽度，然后更新保存每次计算的最大值。直到两根指针相遇可以得到问题的解。

代码如下所示：

1. **int** maxArea(vector<**int**> &height) {
2. **int** n = height.size();
3. **if** (n < 2) {
4. **return** 0;
5. }
6. **int** start = 0;
7. **int** end = height.size()- 1;
8. **int** maxV = INT\_MIN;
9. **while**(start < end) {
10. **int** contain = min(height[end], height[start]) \* (end - start);
11. maxV = max(maxV, contain);
12. **if**(height[start] <= height[end]) {
13. start++;
14. }
15. **else** {
16. end--;
17. }
18. }
19. **return** maxV;
20. }