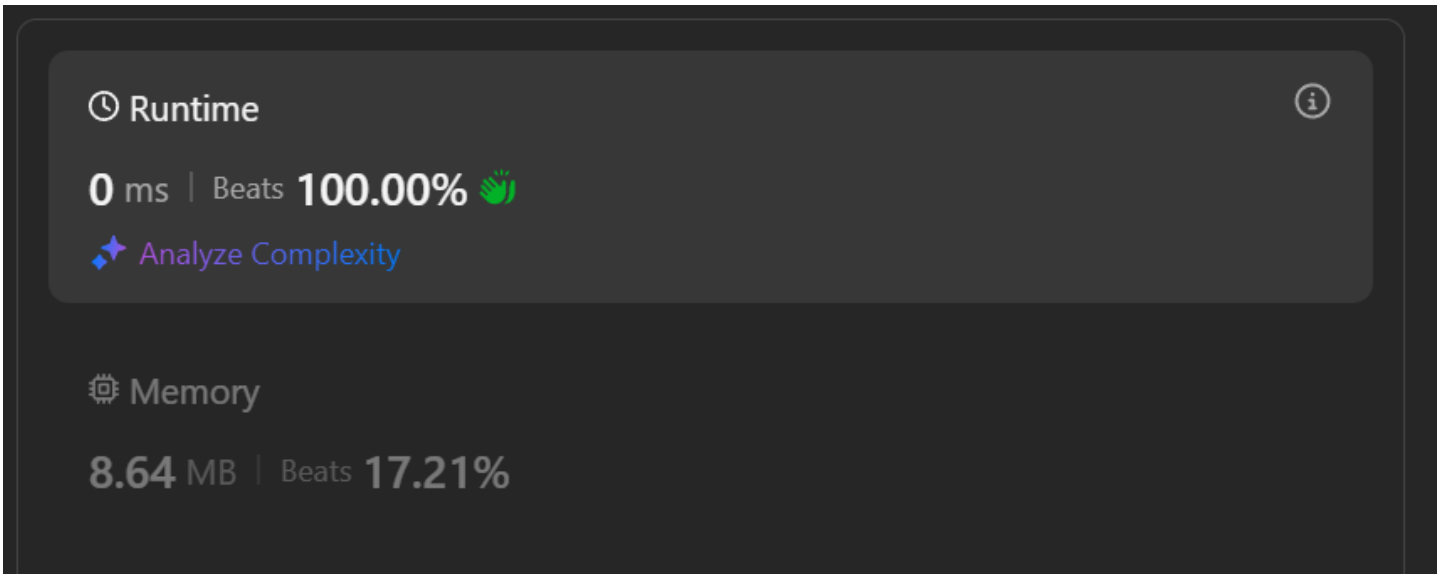
NAME- Akshit Boparai| UID- 22BCS14939 | SECTION- 601/A 1

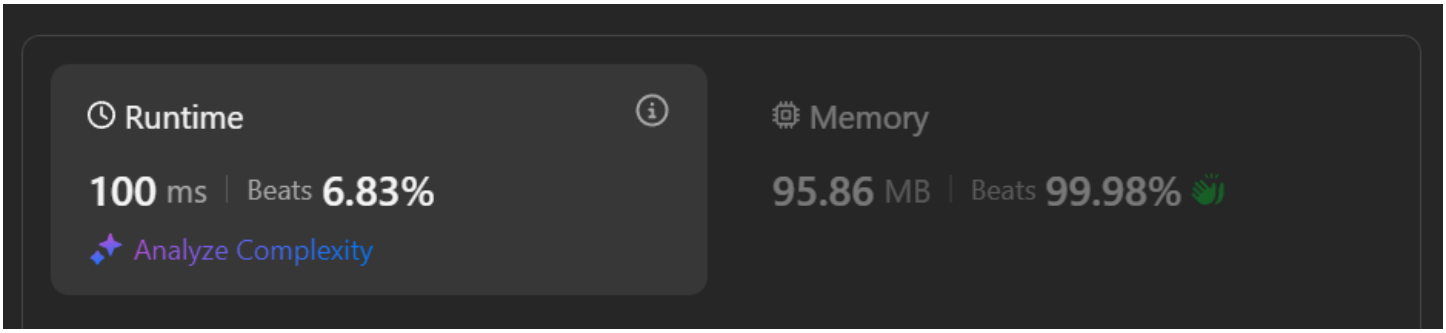
Climbing Stairs

```
class Solution {
public:
    int climbStairs(int n) {
        if (n <= 2) return n;

        vector<int> dp(n + 1, 0);
        dp[1] = 1;
        dp[2] = 2;

        for (int i = 3; i <= n; i++) {
            dp[i] = dp[i - 1] + dp[i - 2];
        }

        return dp[n];
    }
};
```



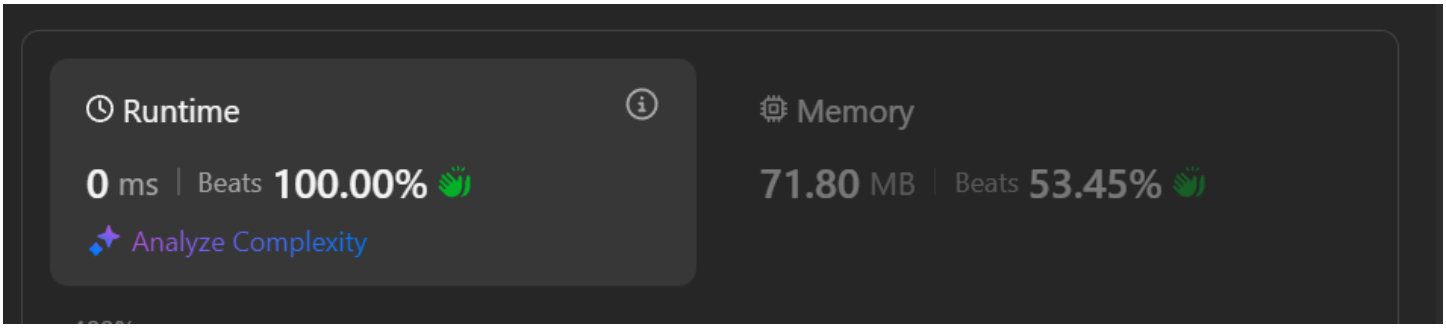
2. Best Time to Buy and Sell a Stock

```
class Solution {
public:
   int maxProfit(vector<int>& arr) {
      int maxPro = 0;
   int n = arr.size();
   int minPrice = INT_MAX;

   for (int i = 0; i < arr.size(); i++) {
      minPrice = min(minPrice, arr[i]);
      maxPro = max(maxPro, arr[i] - minPrice);
   }

   return maxPro;

   }
};
```



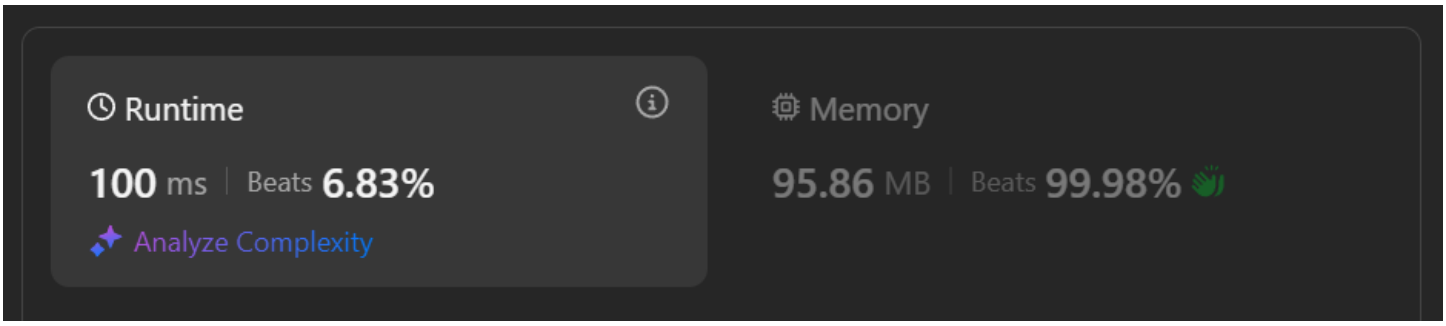
3. Maximum Subarray

```
class Solution {
public:
   int maxSubArray(vector<int>& arr) {
      int n = arr.size();
      long long maxi = LONG_MIN;
      long long sum = 0;

   for (int i = 0; i < n; i++) {

      sum += arr[i];

      if (sum > maxi) {
         maxi = sum;
      }

      if (sum < 0) {
         sum = 0;
      }
   }
  return maxi;
   }
};
```



4. House Robber

```
class Solution {
public:
    int rob(vector<int>& nums) {
        int n = nums.size();

        if (n == 1) {
            return nums[0];
        }

        vector<int> dp(n, 0);

        dp[0] = nums[0];
        dp[1] = max(nums[0], nums[1]);

        for (int i = 2; i < n; i++) {
            dp[i] = max(dp[i - 1], nums[i] + dp[i - 2]);
        }

        return dp[n - 1];
    }
};
```



5. Jump Game

```
class Solution {
public:
    bool canJump(vector<int>& nums) {

        int maxIndex=0;
        for(int i=0;i<nums.size();i++){
            if(i>maxIndex){
                return false;
            }
            maxIndex=max(maxIndex,i+nums[i]);
        }
        return true;
    }
};
```