

Maximum Subarray

```
class Solution {
public:
    int maxSubArray(vector<int>& nums) {
        int curr_max = -1e5, max_sum = nums[0];
        for (auto x : nums) {
            curr_max = max(x, curr_max + x);
            max_sum = max(max_sum, curr_max);
        }
        return max_sum;
    }
};
```

1423. Maximum Points You Can Obtain from Cards

```
class Solution {
public:
    int maxScore(vector<int>& cardPoints, int k) {
        int res = 0;

        //First k elements in our window
        for(int i=0;i<k;i++) res+=cardPoints[i];

        int curr=res;
        for(int i=k-1;i>=0;i--) {
            //We remove the last visited element and add the non-
            //visited element from the last
            curr-=cardPoints[i];
            curr+=cardPoints[cardPoints.size()-k+i];

            //We check the maximum value any possible combination can
            //give
            res = max(res, curr);
        }

        return res;
    }
};
```