HUST

ĐẠI HỌC BÁCH KHOA HÀ NỘI

HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

ONE LOVE. ONE FUTURE.

ĐẠI HỌC
BÁCH KHOA HÀ NỘI
HANOI UNIVERSITY
OF SCIENCE AND TECHNOLOGY

Applied Algorithm Lab

Max even sub-sequence

ONE LOVE. ONE FUTURE.

Max even sub-sequence

- Given a sequence of n integers $a_1, \ldots, a_n$.
- A subsequence of a consists of continuous elements of a

  (for example, ai, ai+1, . . . ,aj).
- The weight of a subsequence is defined to be the sum of its elements. A subsequence is called even-subsequence if its weight is even.
- Find the even-subsequence of a having largest weight.

Input

```
8
4 -5 2 4 -8 2 3 1
```

Output

```
6
```

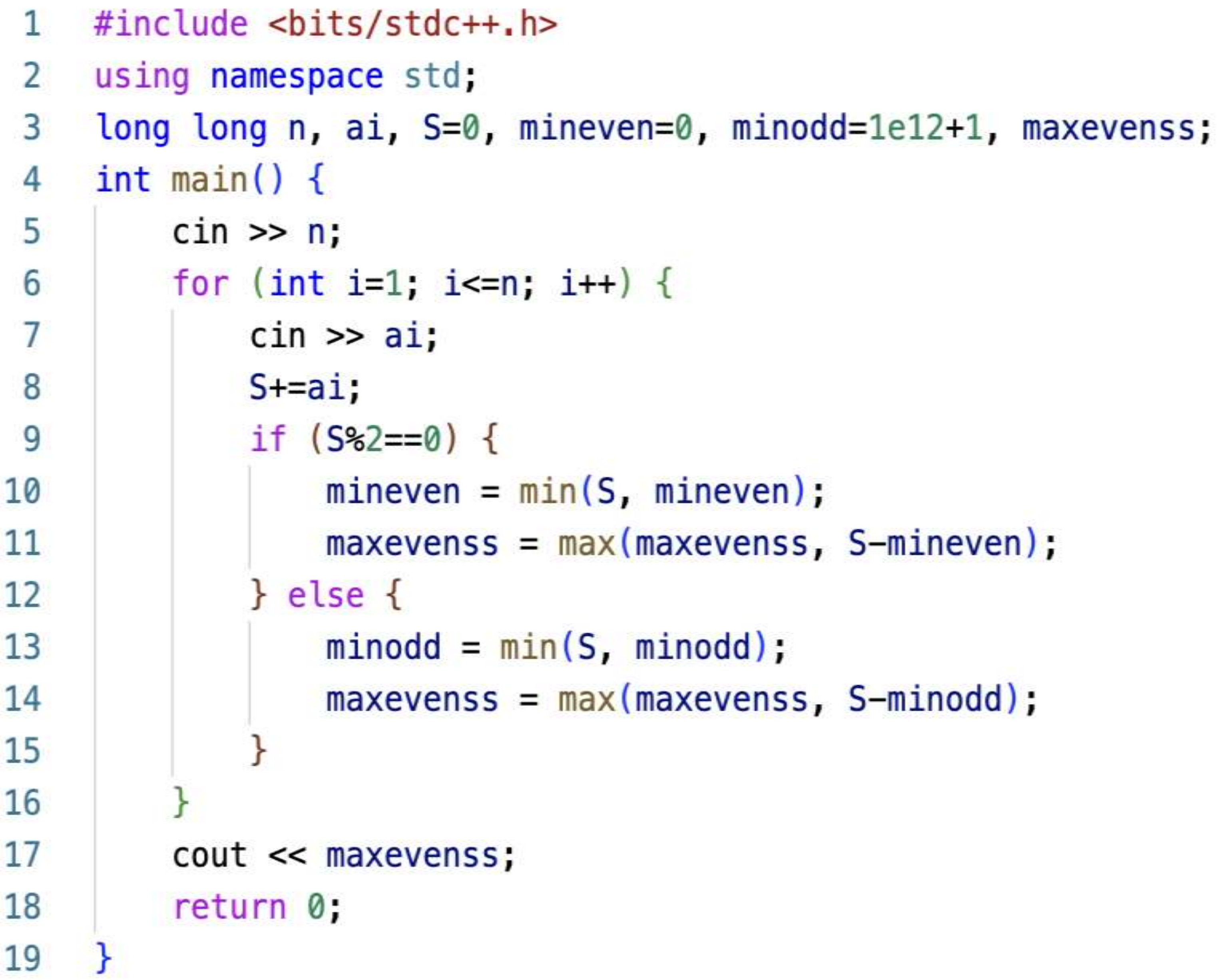
Max even sub-sequence

- Idea to solve: dynamic programming
- Construct a cumulative array S, where S[i] is sum of from a[1] to a[i].
  - Let f[i][0] be minimal S[j] with 1 <= j <= i and j is even.
  - Let f[i][1] be minimal S[j] with 1 <= j <= i and j is odd.
  - Formula:
    - f[i][0] = min(S[i], f[i – 2][0]), with i is even;
    - f[i][1] = min(S[i], f[i – 2][1]), with i is odd;
- Return:

$$\max_{i:1\to n}(S[i] \ - f[i][i \%2])$$

- Complexity: O(n).

Max even sub-sequence - Implementation



```cpp
#include <bits/stdc++.h>
using namespace std;
long long n, ai, S=0, mineven=0, minodd=1e12+1, maxevenss;
int main() {
    cin >> n;
    for (int i=1; i<=n; i++) {
        cin >> ai;
        S+=ai;
        if (S%2==0) {
            mineven = min(S, mineven);
            maxevenss = max(maxevenss, S-mineven);
        } else {
            minodd = min(S, minodd);
            maxevenss = max(maxevenss, S-minodd);
        }
    }
    cout << maxevenss;
    return 0;
}
```

THANK YOU !