Problem 1 – Max Sum

You are given an integer array **arr**, consisting of **N** integers. Find the maximum possible sum of consecutive numbers in **arr**. For example: if the array **arr** consists of the numbers *1, 6, -9, 4, 4, -2, 10, -1,* the maximum possible sum of consecutive numbers is 16 (the consecutive numbers are *4, 4, -2* and *10*)

Your task is to write a JavaScript method named “**Solve**” that solves the problem.

Input

The method **Solve** accepts a zero-based array of strings. Each of the string represents an integer. Element 0 of the array is the number N. Next N elements (from 1 to N) construct the array **arr**.

Output

Your method should return a single number - the maximum possible sum of consecutive numbers.

Example code

```
function Solve(params) {
      var N = parseInt(params[0]);
      var answer = 0;
      // Your code here...
      return answer;
}
```

Constraints

- **N** will be between 1 and 500.
- Each element of **arr** will be between -2 000 000 and +2 000 000.
- Allowed working time for your program: 0.2 seconds. Allowed memory: 16 MB.

Examples (each line represents an element from the only argument of Solve)

Example input	Example output
8 1 6 -9 4 4 -2 10 -1	16

Example input	Example output
6 1 3 -5 8 7 -6	15

Example input	Example output
9 -9 -8 -8 -7 -6 -5 -1 -7 -6	-1