# Problem 1 – Arithmephile +

### You will be given an input of an arbitrary amount of numbers. If a number S is a digit (0 >= S < 10), calculate the product of multiplication of the next S numbers. Find the biggest product among all the S intervals. Note that the intervals may overlap – when you’ve encountered a number that fits the requirement and have calculated the product, the next valid number S may be within this interval.

### Input

* The input data is passed to the first JavaScript function found in your code as an **array of strings** that need to be parsed as numbers.

### Output

* A number, the biggest multiplication should be printed on the console.

### Constraints

* The input may contain up to **10,000** lines (elements)
* The numbers in the input are in range **[-1..10,000] inclusive**
* The numbers denoting ranges (**S**) are in range **[0..9] inclusive**
* Allowed time/memory: 100ms/16MB

**function** *solve*(input) {  
  
 **let** result = 1;  
 **let** set = **new** Set();  
  
 **for** (**let** i = 0; i < input.length; i++) {  
 **let** number = Number(input[i]);  
   
 **if**(number >= 0 && number < 10){  
  
 **for** (**let** j = 1; j <= number; j++) {  
 result \*= input[i+j];  
 }  
 **if**(!isNaN(result)){  
 set.add(result);  
 }  
  
 result = 1;  
 **continue**;  
 }  
 **else** {  
 **continue**;  
 }  
  
 }  
 **let** arr = Array.from(set);  
   
 **let** last = arr.sort(**function** (a, b) {**return** a-b}).pop();  
 **console**.log(last);  
  
}  
  
*solve*([**'100'**, **'200'**, **'2'**, **'3'**, **'2'**, **'3'**, **'2'**, **'1'**, **'1'**]);

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 10  20  2  30  44  3  56  20  24 | 26880 |

|  |  |
| --- | --- |
| **Input** | **Output** |
| 100  200  2  3  2  3  2  1  1 | 12 |