**The task:**

Given the string representations of two integers, return the string representation of the sum of those integers.

A string representation of an integer will contain no characters besides the ten numerals "0" to "9".

/\*\*

 \* @param {string} a - first value

 \* @param {string} b - second value

 \* @return {string} sum of values

 \*/

function sumStrings(a, b)

{

  var carry = 0;

var count = 1;

var sum = '';

var safe\_length = Number.MAX\_SAFE\_INTEGER.toString().length - 1;

var len = a.length || b.length;

var q = Math.ceil(len / safe\_length);

while (q >= count) {

var a\_n = a.slice(-safe\_length \* count, -(count - 1) \* safe\_length != 0 ? -(count - 1) \* safe\_length : undefined);

var b\_n = b.slice(-safe\_length \* count, -(count - 1) \* safe\_length != 0 ? -(count - 1) \* safe\_length : undefined);

if (('' + sum).length > safe\_length) {

carry = Number(('' + sum).slice(0, 1));

sum = sum.slice(1);

}

sum = (+a\_n + +b\_n + +carry) + sum;

count++;

}

return sum;

}

For example:

sumStrings('1','2') // => '3'

sumStrings('800', '9567') // => '10367'

sumStrings('99', '1') // => '100'

sumStrings('00103', '08567') // => '8670'

sumStrings('50095301248058391139327916261', '81055900096023504197206408605') // => '131151201344081895336534324866'