

For this lab, you will do something about: (1) a function with variable-length input arguments, (2) a local function, (3) a little bit of cell arrays, and (4) a little bit of strings.

The goal is to add a list of very large positive integers that are represented as strings. Assume that your main function is named `lab6`. Its usage might look like:

```
s = lab6('1234567890987654321', '121', '99', '100000200000000890');
```

The single return value is a string that represents the correct answer. For the expression above, it is

```
'1244567910987655431'
```

In addition to `lab6`, write a local function within the same m file. Let its name be `lab6worker`. Its task is to add two very large positive integers and return the answer as a string.

Now, what `lab6` needs to do is to loop over the cell array that contains its input strings, call `lab6worker` from inside the loop, and output the final answer.

To convert a string to an array of digits, you can use something like this:

```
'123' -> '0'
```

For this operation, the ASCII codes will be treated as numbers. The result is a double array. How simple! Of course you need to make sure that each character is a digit.

Similarly, to convert an array of digits to a string, you can use something like this:

```
char([1 2 3] + '0')
```

The result is a string. Of course, for the result to be what you expect, you need to first make sure that each array element is an integer in the range of 0 to 9.

Note: You need to do input argument checking in `lab6` to ensure that all the inputs are indeed strings that represent positive integers.

You need to devise the method of the actual addition operation within `lab6worker` yourself. You can use a `while` loop here.