

Client-side input value for test case below

1. character – expects to receive a **fail** code.
 2. one digit – expects to receive a **fail** code.
 3. one digit with operator – expects to receive a **fail** code.
 4. operator with a digit – expects to receive a **fail** code.
 5. invalid operator (%) in between digits with space – expects to receive a **fail** code.
 6. valid operator (+) in between digits with/without space – expects to receive a **success** code.
 7. valid operator (-) in between digits with/without space – expects to receive a **success** code.
 8. valid operator (*) in between digits with/without space – expects to receive a **success** code.
 9. valid operator (/) in between digits with/without space – expects to receive a **success** code.
- * d value will be displayed on the client-side console and the program will be terminated if $d > 2$.

Test cases

1. character – expects to receive a **fail** code.

Client side

Input : test

<<== At Client message to send out: 'test'

Status_Code received : 300

Invalid input!!!

Server side

==>> At Server received message is: 'test'

==>> clientAddress is: 127.0.0.1/64181

<<-- Server send back the result: '-1'

2. one digit – expects to receive a **fail** code.

Client side

<<== At Client message to send out: '2'

Status_Code received : 300

Invalid input!!!

Server side

==>> At Server received message is: '2'

==>> clientAddress is: 127.0.0.1/64182

<<-- Server send back the result: '-1'

3. one digit with operator – expects to receive a **fail** code.

Client side

<<== At Client message to send out: '5+'

Status_Code received : 300

Invalid input!!!

Server side

==>> At Server received message is: '5+'

==>> clientAddress is: 127.0.0.1/64183

<<-- Server send back the result: '-1'

4. operator with a digit – expects to receive a **fail** code.

Client side

<<== At Client message to send out: '* 4'

Status_Code received : 300

Invalid input!!!

Server side

==>> At Server received message is: '* 4'

==>> clientAddress is: 127.0.0.1/64184

<<-- Server send back the result: '-1'

5. invalid operator (%) in between digits with space – expects to receive a **fail** code.

Client side

<<== At Client message to send out: '4 % 2'

Status_Code received : 300

Invalid input!!!

Server side

==>> At Server received message is: '4 % 2'

==>> clientAddress is: 127.0.0.1/64185

<<-- Server send back the result: '-1'

6. valid operator (+) in between digits with/without space – expects to receive a **success** code.

Client side

<<== At Client message to send out: '4 +4'

Status_Code received : 200

Result received! ==>> 4 +4: 8

Server side

==>> At Server received message is: '4 +4'

==>> clientAddress is: 127.0.0.1/64186

<<== Server send back the result: '8'

7. valid operator (-) in between digits with/without space – expects to receive a **success** code.

Client side

<<== At Client message to send out: '4 -4'

Status_Code received : 200

Result received! ==>> 4 -4: 0

Server side

==>> At Server received message is: '4 -4'

==>> clientAddress is: 127.0.0.1/64187

<<== Server send back the result: '0'

8. valid operator (*) in between digits with/without space – expects to receive a **success** code.

Client side

<<== At Client message to send out: '4 *4'

Status_Code received : 200

Result received! ==>> 4 *4: 16

Server side

==>> At Server received message is: '4 *4'

==>> clientAddress is: 127.0.0.1/64188

<<== Server send back the result: '16'

9. valid operator (/) in between digits with/without space – expects to receive a **success** code.

Client side

<<== At Client message to send out: '4 /4'

Status_Code received : 200

Result received! ==>> 4 /4: 1.0

Server side

==>> At Server received message is: '4 /4'

==>> clientAddress is: 127.0.0.1/64189

<<== Server send back the result: '1.0'