# 03. Memory game

Write a program, which receives a sequence of elements. Each element in the sequence will have a twin. Until the player receives "end" from the console, he will receive strings with two integers separated by space, which represent the indexes of elements in the sequence.

If the player tries to cheat and enters two equal indexes or indexes which are out of bounds of the sequence you should add two matching elements in the following format **"-{number of moves until now}a"** at the middle of the sequence and print this message on the console:

**"Invalid input! Adding additional elements to the board"**

### Input

* **On the first line** you will receive **sequence of elements.**

### Output

* Every time the player hit two matching elements you should remove them from the sequence and print on the console the following message:

**"Congrats! You have found matching elements - ${element}!"**

* If the player hit two different elements, you should print on the console the following message:

**"Try again!"**

* If the player hit all matching elements before he receives "end" from the console, you should print on the console the following message:

**"You have won in {number of moves until now} turns!"**

* If the player receives "end" before he hits all matching elements, you should print on the console the following message:

**"Sorry you lose :(**

**{the current sequence's state}"**

## Constraints

* **All elements in the sequence will always have a matching element.**

## Examples

|  |  |
| --- | --- |
| **Input JavaScript** | **Output** |
| [  "1 1 2 2 3 3 4 4 5 5",  "1 0",  "-1 0",  "1 0",  "1 0",  "1 0", "end"] | Congrats! You have found matching elements - 1!  Invalid input! Adding additional elements to the board  Congrats! You have found matching elements - 2!  Congrats! You have found matching elements - 3!  Congrats! You have found matching elements - -1a!  Sorry you lose :(  4 4 5 5 |
| **Comment** | |
| 1)  1 0  1 1 2 2 3 3 4 4 5 5 –> 1 = 1, equal elements, so remove them. Moves: 1  2)  -1 0  -1 is invalid index so we add additional elements  2 2 3 3 -2а -2а 4 4 5 5, Moves: 2  3)  1 0  2 2 3 3 -2а -2а 4 4 5 5 -> 2 = 2, equal elements, so remove them. Moves: 3  4)  1 0  3 3 -2а -2а 4 4 5 5 -> 3 = 3, equal elements, so remove them. Moves: 4  5)  1 0  -2а -2а 4 4 5 5 -> -2а = -2а, equal elements, so remove them. Moves: 5  6)  We receive end command.  There are still elements in the sequence, so we loose the game.  Final state - 4 4 5 5 | |
| [  "a 2 4 a 2 4",  "0 3",  "0 2","0 1","0 1", "end"] | Congrats! You have found matching elements - a!  Congrats! You have found matching elements - 2!  Congrats! You have found matching elements - 4!  You have won in 3 turns! |
| [  "a 2 4 a 2 4",  "4 0",  "0 2",  "0 1","0 1", "end"] | Try again!  Try again!  Try again!  Try again!  Sorry you lose :(  a 2 4 a 2 4 |

function memoryGame(arr) {

    arr.pop()

    let board = arr.shift().split(' ')

    let moves = 0

    while (board.length > 0 &&  arr.length > 0) {

      moves++

      const [fIndex, sIndex] = arr.shift().split(' ')

      let [firstEl, secondEl] = [board[fIndex], board[sIndex]]

      if (firstEl == undefined || secondEl == undefined) {

        board.splice(Math.floor(board.length / 2), 0, `-${moves}a`, `-${moves}a`)

        console.log('Invalid input! Adding additional elements to the board')

      } else if (firstEl == secondEl) {

        board = board.filter((el, i) => i == fIndex || i == sIndex ? null : el)

        console.log(`Congrats! You have found matching elements - ${firstEl}!`)

      } else {

        console.log('Try again!')

      }

    }

    return board.length > 0 ? `Sorry you lose :(\n${board.join(' ')}` : `You have won in ${moves} turns!`

  }

  console.log(memoryGame([`a 2 4 a 2 4`, `0 3`, `0 2`, `0 1`, `0 1`, `end`]))