function decode(expr) {

let str = expr.match(/.{1,2}/g);

let rez = str.slice();

for (let i = 0; i < str.length; i++) {

if (str[i] === "00") {

str.splice(i , 1);

}

if (str[i] === "10") {

str.splice(i, 1, ".");

}

if (str[i] === "11") {

str.splice(i, 1, "-");

}

}

console.log(rez);

console.log(str);

}

decode(expr = "00101010100000000010001011101000101110100000111111")

const MORSE\_TABLE = {

".-": "a",

"-...": "b",

"-.-.": "c",

"-..": "d",

".": "e",

"..-.": "f",

"--.": "g",

"....": "h",

"..": "i",

".---": "j",

"-.-": "k",

".-..": "l",

"--": "m",

"-.": "n",

"---": "o",

".--.": "p",

"--.-": "q",

".-.": "r",

"...": "s",

"-": "t",

"..-": "u",

"...-": "v",

".--": "w",

"-..-": "x",

"-.--": "y",

"--..": "z",

".----": "1",

"..---": "2",

"...--": "3",

"....-": "4",

".....": "5",

"-....": "6",

"--...": "7",

"---..": "8",

"----.": "9",

"-----": "0",

};

function decode(expr) {

let str = expr.match(/.{1,2}/g);

for (let i = 0; i < str.length; i++) {

if (str[i] === "00") {

str.splice(i , 1);

}

if (str[i] === "10") {

str.splice(i, 1, ".");

}

if (str[i] === "11") {

str.splice(i, 1, "-");

}

}

console.log(str.split(" ")

.map((word) =>

word

.split(" ")

.map((character) => MORSE\_TABLE[character])

.join("")

)

.join(" ")

.trim());

}

decode(expr = "00101010100000000010001011101000101110100000111111\*\*\*\*\*\*\*\*\*\*00001011110000111111000010111000101110100000111010")

str.splice(str.indexOf(“0”));

console.log(str.splice(index).join(""))

let messageConverted = [];

str.split(" ").map(function (word) {

word.split(" ").map(function (letter) {

messageConverted.push(MORSE\_TABLE [letter]);

});

messageConverted.push(" ");

});

console.log(messageConverted.join(""));

var ind=0;

var answer = [];

const moerse\_keys = Object.keys(MORSE\_TABLE);

const moerse\_values = Object.values(MORSE\_TABLE);

var words\_in\_morse = morseCode.split (' ');

var letters\_in\_morse =

words\_in\_morse.map(word => word.split(' '));

for (let i=0 ; i<letters\_in\_morse.length ; i++) {

for (let j=0 ; j<letters\_in\_morse[i].length ; j++) {

if ( moerse\_keys.includes(letters\_in\_morse[i][j]) ) {

ind = moerse\_keys.indexOf( letters\_in\_morse[i][j] );

answer.push(moerse\_values[ind]);

}

if (j===letters\_in\_morse[i].length-1 ) {

answer.push(' ');

}

}

}

answer.pop();

console.log(answer.join(''))

"......-...-..--- .-----.-..-..-.."

return expr

.split(' ')

.map(

a => a

.split(' ')

.map(

b => MORSE\_TABLE [b]

).join('')

).join(' ');

function decode(expr) {

const MORSE\_TABLE = {

".-": "a",

"-...": "b",

"-.-.": "c",

"-..": "d",

".": "e",

"..-.": "f",

"--.": "g",

"....": "h",

"..": "i",

".---": "j",

"-.-": "k",

".-..": "l",

"--": "m",

"-.": "n",

"---": "o",

".--.": "p",

"--.-": "q",

".-.": "r",

"...": "s",

"-": "t",

"..-": "u",

"...-": "v",

".--": "w",

"-..-": "x",

"-.--": "y",

"--..": "z",

".----": "1",

"..---": "2",

"...--": "3",

"....-": "4",

".....": "5",

"-....": "6",

"--...": "7",

"---..": "8",

"----.": "9",

"-----": "0",

};

let str = expr.match(/.{1,2}/g);

let rez = str.slice();

for (let i = 0; i < str.length; i++) {

if (str[i] === "00") {

str.splice(i , 1);

i--;

}

if (str[i] === "10") {

str.splice(i, 1, ".");

}

if (str[i] === "11") {

str.splice(i, 1, "-");

}

if (str[i] === "\*\*") {

str.splice(i, 5, " ");

}

}

str = str.join("");

var ind=0;

var answer = [];

const moerse\_keys = Object.keys(MORSE\_TABLE);

const moerse\_values = Object.values(MORSE\_TABLE);

var words\_in\_morse = str.split (' ');

var letters\_in\_morse =

words\_in\_morse.map(word => word.split(' '));

for (let i=0 ; i<letters\_in\_morse.length ; i++) {

for (let j=0 ; j<letters\_in\_morse[i].length ; j++) {

if ( moerse\_keys.includes(letters\_in\_morse[i][j]) ) {

ind = moerse\_keys.indexOf( letters\_in\_morse[i][j] );

answer.push(moerse\_values[ind]);

}

if (j===letters\_in\_morse[i].length-1 ) {

answer.push(' ');

}

}

}

answer.pop();

console.log(answer.join(''))

}

decode(expr = "00101010100000000010001011101000101110100000111111\*\*\*\*\*\*\*\*\*\*00001011110000111111000010111000101110100000111010")

function decode(expr) {

const MORSE\_TABLE = {

".-": "a",

"-...": "b",

"-.-.": "c",

"-..": "d",

".": "e",

"..-.": "f",

"--.": "g",

"....": "h",

"..": "i",

".---": "j",

"-.-": "k",

".-..": "l",

"--": "m",

"-.": "n",

"---": "o",

".--.": "p",

"--.-": "q",

".-.": "r",

"...": "s",

"-": "t",

"..-": "u",

"...-": "v",

".--": "w",

"-..-": "x",

"-.--": "y",

"--..": "z",

".----": "1",

"..---": "2",

"...--": "3",

"....-": "4",

".....": "5",

"-....": "6",

"--...": "7",

"---..": "8",

"----.": "9",

"-----": "0",

};

let str = expr.match(/.{1,10}/g);

for (let i = 0; i < str.length; i++) {

let letter = str[i].split("00").join("");

console.log(letter);

}

/\*for (let i = 0; i < str.length; i++) {

if (str[i] === "00") {

str.splice(i , 1);

i--;

}

if (str[i] === "10") {

str.splice(i, 1, ".");

}

if (str[i] === "11") {

str.splice(i, 1, "-");

}

if (str[i] === "\*\*") {

str.splice(i, 5, " ");

}

}\*/

// str = str.join("");

// console.log(expr

// .split(' ')

// .map(

// a => a

// .split(' ')

// .map(

// b => MORSE\_TABLE [b]

// ).join('')

// ).join(' '));

}

decode(expr = "00101010100000000010001011101000101110100000111111\*\*\*\*\*\*\*\*\*\*00001011110000111111000010111000101110100000111010")