

Ejercicio 1

```
let w;
let tam, longi, k = 0;
let sigma = [];

const pedirDatos = () => {
  tam = prompt("Digite el tamaño del alfabeto");
  for (let i = 0; i < tam; i++) {
    sigma.push(prompt(`${(i + 1)}.- Digite la letra`));
  }
  w = prompt("Digite la palabra");
  w = w.split('');
  console.log(w);
}

const palindromo = (sigma, w) => {
  for (let i = 0; i < sigma.length; i++) {
    let elemento = sigma[i];
    if (w.includes(elemento)){
      k++;
    }
  }
  let p = w.reverse();
  console.log(p);
  if (k == w.length && w == p) {
    document.write("La palabra si tiene los simbolos del alfabeto y es un palindromo");
  }
  if(k == w.length) {
    document.write("La palabra si tiene los simbolos del alfabeto pero no es un palindromo");
  }
  if (w == p) {
    document.write("La palabra si es un palindromo pero algun simbolo no esta en el alfabeto");
  }
}

pedirDatos();
palindromo(sigma, w);
```

La palabra si es un palindromo pero algun simbolo no esta en el alfabeto

Ejercicio 2

```
let tam, longi;
let lista = [];
let nuevaLista = [];
let w = [];
let a, b = [];

const pedirDatos = () => {
  tam = prompt("Digite el tamaño del lenguaje");
  for (let i = 0; i < tam; i++) {
    lista.push(prompt(`${(i + 1)}.- Digite la letra`));
  }
  longi = prompt("Digite la longitud del conjunto");
}

const lenguaje = (letra) =>{
  if(letra.length > longi){
    return;
  }
  let letra2 = letra;
  nuevaLista.push(letra2);
  for(let i = 0; i < tam; i++){
    lenguaje(letra2 + lista[i]);
  }
}

const bucle = () =>{
  for (let i = 0; i < lista.length; i++) {
    lenguaje(lista[i]);
  }
}

const mostrar = () => {
  document.write(`^<br>`);
  for (const iterator of nuevaLista) {
    document.write(`${iterator} <br>`);
  }
}

const arregloPar = () => {
```

```
    for (let i = 0; i < lista.length; i++) {  
        a = lista[i];  
        a += a;  
        b.push(a);  
    }  
}  
  
const palabrasPar = () => {  
    let j = 0;  
    for (let i = 0; i < nuevaLista.length; i++) {  
        for(let j = 0; j < b.length; j++) {  
            let x = b[j];  
            if(nuevaLista.includes(x)) {  
                console.log(nuevaLista.includes(x));  
            }  
        }  
    }  
}  
}  
  
pedirDatos();  
bucle();  
mostrar();  
  
// arregloPar();  
// palabrasPar();
```

^

a

aa

aaa

aaaa

aaab

aaac

aab

aaba

aabb

aabc

aac

aaca

aacb

aacc

ab

aba

abaa

abab

abac

abb

abba

abbb

abbc

abc

abca

abcb

abcc

ac

aca

acaa

acab

acac

acb

acba

acbb

acbc

acc

acca

accb

accc

b
