

## Actividad 6 Javascript

### **Enunciado del Ejercicio: Encriptación de Mensaje Personalizada**

Desarrolla un programa que solicite al usuario una **cadena de texto** (mensaje) y lo **encripte** siguiendo las siguientes reglas:

---

#### **Reglas de Encriptación**

##### 1. Consonantes:

- Cada consonante debe ser desplazada **9 posiciones hacia adelante** en el alfabeto.
- Si al avanzar se pasa de la 'z', debe **volver al inicio del alfabeto** (comportamiento circular).

##### 2. Vocales (a, e, i, o, u):

- Cada vocal debe desplazarse **5 posiciones hacia atrás** en el alfabeto.

##### 3. Excepciones

**especiales:**

Algunas letras tienen reglas únicas que se deben aplicar **antes** de lo anterior:

Letra	Acción
c	Mover 1 posición hacia atrás
o	Mover 1 posición hacia atrás
d	Mover 3 posiciones hacia atrás
e	Mover 4 posiciones hacia atrás

##### 4. Regla de revertir:

- Si después del desplazamiento una letra se convierte en 'c', 'o', 'd' o 'e', **reviértela a su valor original**.
- 

#### **Restricciones del mensaje**

- El mensaje debe estar **en minúsculas**.
  - No puede contener espacios ni caracteres especiales (solo letras).
  - No puede estar vacío.
- 

#### **Objetivo**

Al final, el programa debe mostrar el mensaje **encriptado** según todas las reglas anteriores.

---



## Orientación para resolverlo paso a paso

### 1. Validar el mensaje

- Asegúrate de que no esté vacío.
- Verifica que solo contenga letras minúsculas (puedes usar expresiones regulares o funciones de validación).

### 2. Recorrer cada letra del mensaje

- Para cada letra, determina si es vocal, consonante o una excepción.

### 3. Aplicar reglas especiales

- Antes de hacer cualquier desplazamiento, revisa si es una letra con excepción (c, o, d, e) y aplica su movimiento especial.

### 4. Aplicar desplazamientos

- Para vocales: mover 5 posiciones hacia atrás.
- Para consonantes: mover 9 posiciones hacia adelante.
- Usa funciones con código ASCII (`charCodeAt` y `fromCharCode`) para calcular el desplazamiento.

### 5. Aplicar la regla de reversión

- Si el resultado de la transformación da como resultado c, o, d, o e, reemplaza esa letra por la **original**.

### 6. Unir todas las letras transformadas

- Guarda cada letra modificada en una nueva cadena de texto.

### 7. Mostrar el resultado final al usuario



## Tip extra: Rotación circular en el alfabeto

Usa esta fórmula para desplazamientos circulares:

```
// letra es una minúscula
// desplazamiento puede ser positivo o negativo
function moverLetra(letra, desplazamiento) {
  const codigoA = 'a'.charCodeAt(0);
  return String.fromCharCode(((letra.charCodeAt(0) - codigoA + desplazamiento + 26) % 26) + codigoA);
}
```