

„multicol

## Informe de Laboratorio

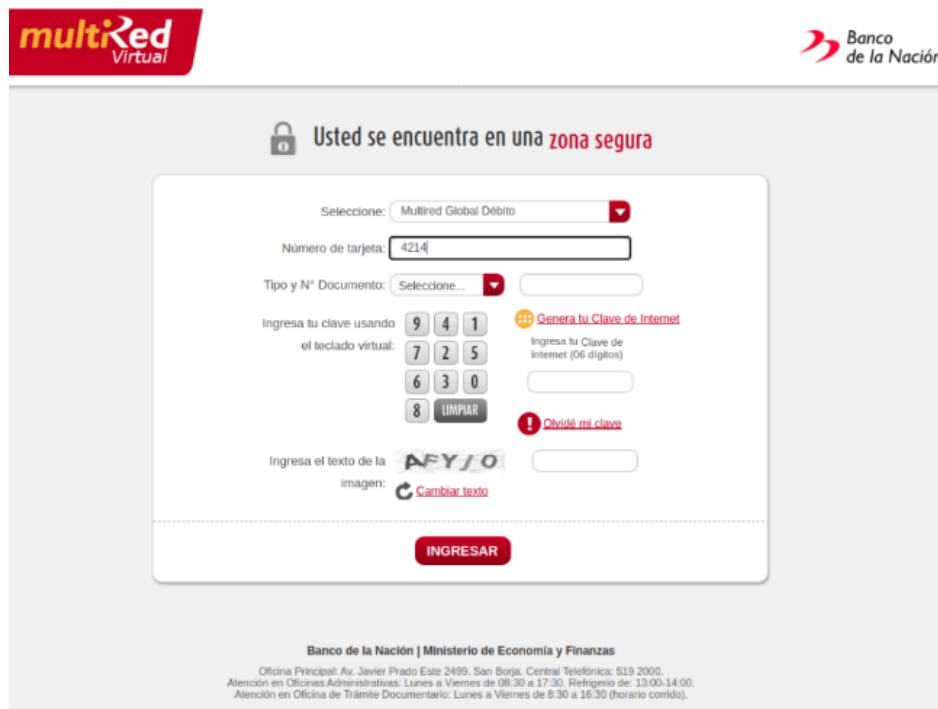
### Tema: JavaScript

Nota

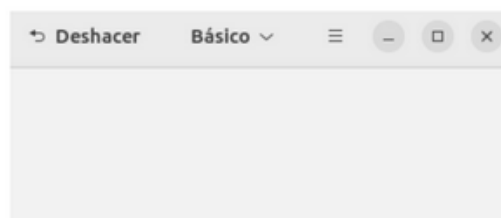
Guillermo Ampuero, gampuero@ulasalle.edu.pe

### 1. Tarea

- Ejercicio 01: Cree un teclado random para banca por internet.



- Ejercicio 02: Cree una calculadora básica como la de los sistemas operativos, que pueda utilizar la función eval() y que guarde todas las operaciones en una pila. Mostrar la pila al pie de la página web.



## 2. Ejercicio 01

### 2.1. Código

#### 2.1.1. ejercicio01.html

Listing 1: Html

```
<!DOCTYPE html> <!-- Declaracin del tipo de documento -->
<html lang="es"> <!-- Inicio de la etiqueta html con el atributo lang para especificar el
    idioma -->
<head>
<meta charset="UTF-8"> <!-- Especificacin de la codificacin de caracteres -->
<title>Banco de la Nacin - Multired Virtual</title> <!-- Ttulo de la pgina -->
<link rel="stylesheet" href="css/css_ejercicio_01.css"> <!-- Enlace al archivo CSS externo
-->
<script src="js/script_ejercicio_01.js" defer></script> <!-- Enlace al archivo JavaScript
    externo con el atributo defer para carga asncrona -->
</head>
<body>
<div id="contenedor"> <!-- Div contenedor principal -->
    <div id="cabecera"> <!-- Cabecera de la pgina -->
         <!-- Imagen del
            logotipo de Multired con atributo alt para accesibilidad -->
         <!-- Imagen del
            logotipo del Banco de la Nacin con atributo alt para accesibilidad -->
    </div>
    <div id="cuerpo"> <!-- Cuerpo de la pgina -->
        <h1>Usted se encuentra en una <span>zona segura</span></h1> <!-- Ttulo principal -->
        <div id="login"> <!-- Formulario de inicio de sesin -->
            <form id="form" name="form"> <!-- Formulario con identificador form y nombre form
            -->
                <select id="cboTipoTarjeta" name="cboTipoTarjeta"
                    onchange="cambiarTipoTarjeta();" <!-- Selector para tipo de tarjeta -->
                    <option value='02'>Multired Global Dbito</option> <!-- Opcin de tarjeta
                        Multired Global Dbito -->
                    <option value='01'>DNI (Cuenta Corriente)</option> <!-- Opcin de tarjeta
                        DNI (Cuenta Corriente) -->
                </select>
                <input type="text" name="txtNumeroTarjeta" id="txtNumeroTarjeta"
                    maxlength="16"> <!-- Campo de entrada para nmero de tarjeta -->
                <input type="text" name="txtDNI" id="txtDNI" class="tarjeta_dni" maxlength="8"
                    style="display:none;" <!-- Campo de entrada para DNI, oculto por defecto
                    -->
                <select id="cboTipoDoc" name="cboTipoDoc" onchange="limpiarNumDoc();"
                    style="display: none;" <!-- Selector para tipo de documento, oculto por
                    defecto -->
                    <option value="">Seleccione...</option> <!-- Opcin predeterminada -->
                    <option value="1">DNI</option> <!-- Opcin DNI -->
                    <option value="7">Pasaporte</option> <!-- Opcin Pasaporte -->
                    <option value="3">C. Extranjeria</option> <!-- Opcin Carn de Extranjera -->
                    <option value="6">RUC</option> <!-- Opcin RUC -->
                </select>
                <input type="text" class="input-chico3" name="txtNumDoc" id="txtNumDoc"
                    maxlength="12"> <!-- Campo de entrada para nmero de documento -->
            </form>
        </div>
    </div>
</div>
```

```
<input type="password" name="txtPassword" id="txtPassword" maxlength="6"
      readonly="readonly"> <!-- Campo de entrada para contrasea, de solo lectura
-->
<input type="text" name="txtCaptcha" id="txtCaptcha" maxlength="5"> <!-- Campo
de entrada para captcha -->
<input type="button" value="INGRESAR" onclick="Autenticar();"> <!-- Botn de
inicio de sesin -->
</form>
</div>
</div>
<div id="pie-pagina"> <!-- Pie de pgina -->
  <div id="titulo-pie-pagina">Banco de la Nacin | Ministerio de Economa y
  Finanzas</div> <!-- Ttulo del pie de pgina -->
  <div id="oficinas"> <!-- Informacin de las oficinas -->
    <p>Oficina Principal: Av. Javier Prado Este 2499. San Borja. Central Telefnica:
    519 2000.</p> <!-- Direccin y telefono de la oficina principal -->
    <p>Atencin en Oficinas Administrativas: Lunes a Viernes de 08:30 a 17:30.
    Refrigerio de: 13:00-14:00.</p> <!-- Horario de atencin en oficinas
administrativas -->
    <p>Atencin en Oficina de Trmite Documentario: Lunes a Viernes de 8:30 a 16:30
    (horario corrido).</p> <!-- Horario de atencin en la oficina de trmite
documentario -->
  </div>
</div>
</div>
</body>
</html>
```

### 2.1.2. script\_ejercicio\_01

Listing 2: JS

```
function generarNumerosAleatorios() {
// Array con los nmeros del 0 al 9
var numeros = ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9'];

// Obtener el contenedor de los botones
var contenedorBotones = document.getElementById("botones-clave");

// Array para almacenar los botones
var botones = contenedorBotones.querySelectorAll(".boton-clave");

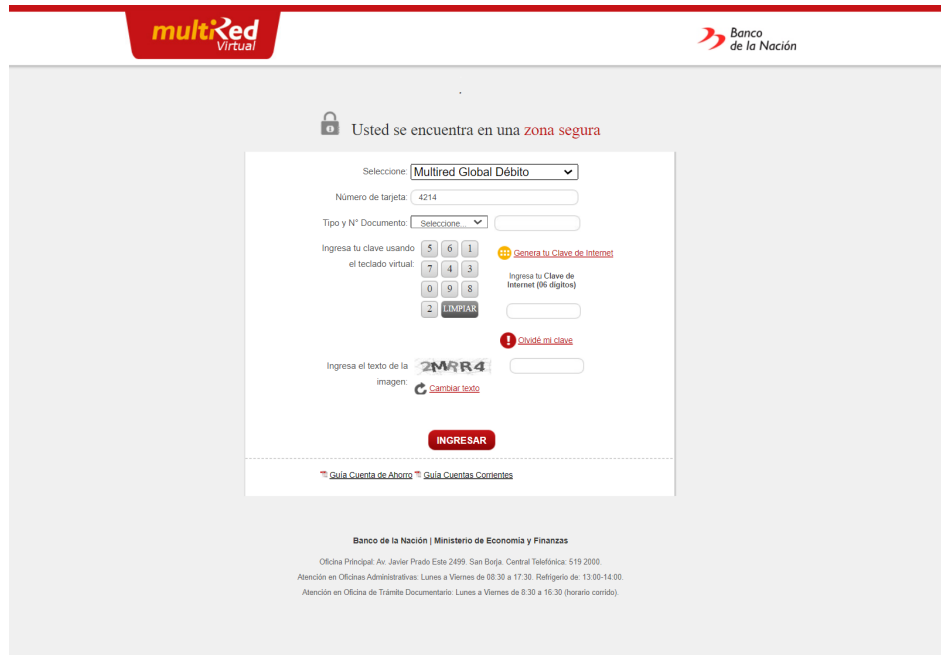
// Array para almacenar los nmeros generados aleatoriamente
var numerosAleatorios = [];

// Generar nmeros aleatorios y asignarlos a los botones
for (var i = 0; i < botones.length - 1; i++) { // Excluimos el ltimo botn (limpiar)
  // Generar un ndice aleatorio dentro del rango del array de nmeros
  var randomIndex = Math.floor(Math.random() * numeros.length);
  // Eliminar el nmero aleatorio del array de nmeros y almacenarlo en una variable
  var numeroAleatorio = numeros.splice(randomIndex, 1)[0];
  // Asignar el nmero aleatorio como el contenido de texto del botn
  botones[i].textContent = numeroAleatorio;
  // Establecer un atributo onclick en el botn para llamar a una funcin con el nmero
  aleatorio como argumento
```

```
botones[i].setAttribute("onclick", "evalRanTable(' + numeroAleatorio + "')");
}
}
```

## 2.2. Resultados

### 2.2.1.



En la imagen se puede visualizar la página de la entidad bancaria con su respectivo teclado.

## 3. Ejercicio 02

### 3.1. Código

#### 3.1.1. ejercicio01.html

Listing 3: Html

```
<!DOCTYPE html> <!-- Declaración del tipo de documento como HTML5 -->
<html lang="en"> <!-- Declaración del idioma principal de la página, en este caso, inglés (en) -->
<head>
  <meta charset="UTF-8"> <!-- Especificación de la codificación de caracteres UTF-8 -->
  <meta name="viewport" content="width=device-width, initial-scale=1.0"> <!-- Configuración de la vista adaptable para dispositivos móviles -->
  <title>Calculadora con Pila</title> <!-- Título de la página -->
  <link rel="stylesheet" href="css/css_ejercicio_02.css"> <!-- Enlace al archivo CSS para estilizar la página -->
</head>
<body>
```

```
<div class="calculator"> <!-- Div para la calculadora -->
  <input type="text" id="display" readonly> <!-- Campo de texto de entrada que muestra los
    nmeros y operaciones -->
  <div class="buttons"> <!-- Div para los botones de la calculadora -->
    <button onclick="addToDisplay('(')"></button> <!-- Botn para agregar el parntesis de
      apertura '(' -->
    <button onclick="addToDisplay(')')"></button> <!-- Botn para agregar el parntesis de
      cierre ')' -->
    <button onclick="addToDisplay('x')">x</button> <!-- Botn para agregar el operador de
      multiplicacin 'x' -->
    <button onclick="clearDisplay()">Clear</button> <!-- Botn para limpiar el campo de
      texto -->
    <!-- Botones para los dgitos del 0 al 9 y los operadores +, -, *, / y el punto decimal
      -->
    <button onclick="addToDisplay('7')">7</button>
    <button onclick="addToDisplay('8')">8</button>
    <button onclick="addToDisplay('9')">9</button>
    <button onclick="addToDisplay('/')">/</button>
    <button onclick="addToDisplay('4')">4</button>
    <button onclick="addToDisplay('5')">5</button>
    <button onclick="addToDisplay('6')">6</button>
    <button onclick="addToDisplay('*')">*</button>
    <button onclick="addToDisplay('1')">1</button>
    <button onclick="addToDisplay('2')">2</button>
    <button onclick="addToDisplay('3')">3</button>
    <button onclick="addToDisplay('-')">-</button>
    <button onclick="addToDisplay('0')">0</button>
    <button onclick="addToDisplay('.')">.</button>
    <button onclick="addToDisplay('+')">+</button>
    <button onclick="addToDisplay('=')">=</button> <!-- Botn de igual para calcular el
      resultado -->
    <button onclick="calculate()">Calcular</button> <!-- Botn para iniciar el clculo -->
  </div>
</div>
<div class="stack" id="stack"> <!-- Div para la pila de operaciones -->
  <h2>Pila de operaciones:</h2> <!-- Encabezado para la pila de operaciones -->
  <ul id="stack-list"></ul> <!-- Lista no ordenada donde se mostrarn las operaciones
    ingresadas -->
</div>
<script src="js/script_ejercicio_02.js"></script> <!-- Enlace al archivo JavaScript para el
  funcionamiento de la calculadora -->
</body>
</html>
```

### 3.1.2. script\_ejercicio\_02

Listing 4: JS

```
let displayValue = ''; // Almacena el valor mostrado en el campo de texto de la
  calculadora
let stack = []; // Almacena las operaciones realizadas para mostrar en la pila

// Funcion para calcular el resultado de la operacion ingresada
function calculate() {
```

```
const equation = displayValue; // Obtiene la operacion ingresada
let result;

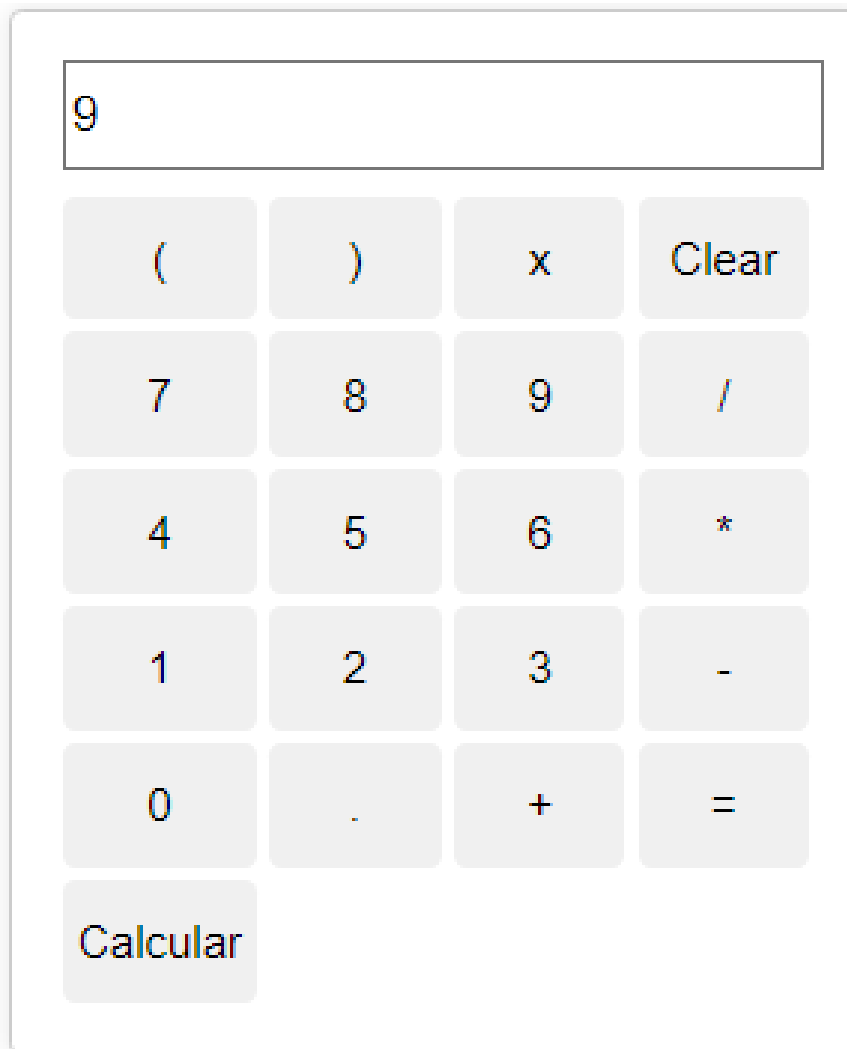
// Verificar si es una ecuacion o una asignacion
if (equation.includes('=')) {
  const parts = equation.split('='); // Divide la operacion en dos partes: variable y
  // expresion
  const variable = parts[0].trim(); // Obtiene la variable de la asignacion
  const expression = parts[1].trim(); // Obtiene la expresion de la asignacion
  result = eval(expression); // Evalua la expresion para obtener el resultado
  stack.push({ operation: equation, result: result, variable: variable }); // Agrega la
  // operacion a la pila
} else {
  result = eval(equation); // Evalua la operacion para obtener el resultado
  stack.push({ operation: equation, result: result }); // Agrega la operacion a la pila
}

displayValue = ''; // Reinicia el valor del campo de texto
document.getElementById('display').value = result; // Muestra el resultado en el campo de
// texto
updateStack(); // Actualiza la lista de la pila de operaciones
}

// Funcion para actualizar la lista de la pila de operaciones
function updateStack() {
  const stackList = document.getElementById('stack-list'); // Obtiene la lista de la pila de
  // operaciones
  stackList.innerHTML = ''; // Limpia el contenido actual de la lista

  // Itera sobre cada operacion en la pila
  stack.forEach((item, index) => {
    const li = document.createElement('li'); // Crea un elemento de lista
    if (item.variable) {
      li.textContent = `${item.variable} = ${item.result}`; // Muestra la asignacion de
      // variable
    } else {
      li.textContent = `${item.operation} = ${item.result}`; // Muestra la operacion y su
      // resultado
    }
    li.onclick = () => {
      displayValue = stack[index].operation; // Restaura la operacion seleccionada en el
      // campo de texto
      document.getElementById('display').value = displayValue; // Actualiza el valor mostrado
      // en el campo de texto
    };
    stackList.appendChild(li); // Agrega el elemento de lista a la lista de la pila de
    // operaciones
  });
}
```

### 3.2. Resultados



## Pila de operaciones:

$$2 * 2 = 4$$

$$x = 9$$

En la imagen podemos visualizar la calculadora, con la pila de operaciones.

## 4. Ejercicio 03

### 4.1. Código

#### 4.1.1. Html

Listing 5: Html

```
<!DOCTYPE html> <!-- Declaracion del tipo de documento como HTML5 -->
<html lang="es"> <!-- Declaracion del idioma principal de la pagina, en este caso, espaol
(es) -->
<head>
  <meta charset="UTF-8"> <!-- Especificacion de la codificacion de caracteres UTF-8 -->
  <meta name="viewport" content="width=device-width, initial-scale=1.0"> <!-- Configuracion
de la vista adaptable para dispositivos moviles -->
  <link rel="stylesheet" href="css/css_ejercicio_03.css"> <!-- Enlace al archivo CSS para
estilizar la pagina -->
  <title>Ahorcado</title> <!-- Titulo de la pagina -->
</head>
<body>
  <canvas id="canvas" width="400" height="400"></canvas><br> <!-- Elemento lienzo para
dibujar el ahorcado -->
  <button onclick="dibujarAhorcado()">Generar Ahorcado</button> <!-- Boton para generar el
dibujo del ahorcado -->

  <script src="js/script_ejercicio_03.js"></script> <!-- Enlace al archivo JavaScript para el
funcionamiento del ahorcado -->
</body>
</html>
```

#### 4.1.2. JS

Listing 6: JS

```
const canvas = document.getElementById('canvas'); // Obtiene el elemento canvas del
HTML
const pizarra = canvas.getContext('2d'); // Obtiene el contexto de dibujo 2D del canvas

let partesDibujadas = 0; // Contador para las partes del ahorcado dibujadas

function dibujarEstructura() {
  // Dibujar poste vertical
  pizarra.beginPath();
  pizarra.moveTo(100, 300);
  pizarra.lineTo(100, 50);
  pizarra.stroke();

  // Dibujar poste horizontal
  pizarra.moveTo(50, 300);
  pizarra.lineTo(200, 300);
  pizarra.stroke();

  // Dibujar barra horizontal superior
  pizarra.moveTo(100, 50);
  pizarra.lineTo(250, 50);
```



```
pizarra.stroke();

// Dibujar barra vertical para sostener la cuerda
pizarra.moveTo(250, 50);
pizarra.lineTo(250, 100);
pizarra.stroke();
}

function dibujarCabeza() {
    pizarra.beginPath();
    pizarra.arc(250, 130, 30, 0, Math.PI * 2);
    pizarra.stroke();
}

function dibujarCuerpo() {
    pizarra.moveTo(250, 160);
    pizarra.lineTo(250, 230);
    pizarra.stroke();
}

function dibujarBrazoIzquierdo() {
    pizarra.moveTo(250, 180);
    pizarra.lineTo(220, 200);
    pizarra.stroke();
}

function dibujarBrazoDerecho() {
    pizarra.moveTo(250, 180);
    pizarra.lineTo(280, 200);
    pizarra.stroke();
}

function dibujarPiernaIzquierda() {
    pizarra.moveTo(250, 230);
    pizarra.lineTo(220, 270);
    pizarra.stroke();
}

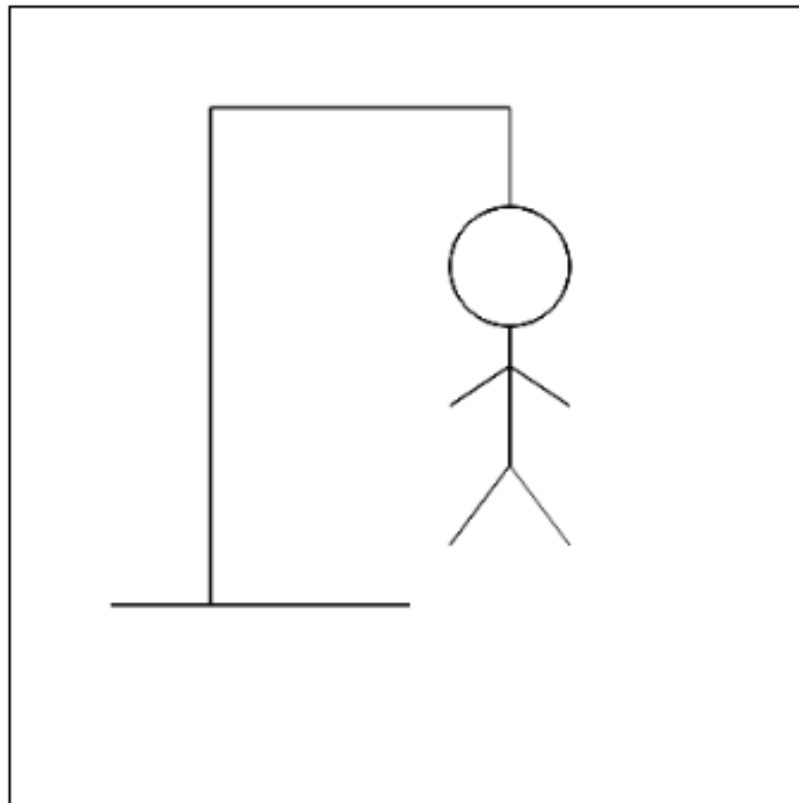
function dibujarPiernaDerecha() {
    pizarra.moveTo(250, 230);
    pizarra.lineTo(280, 270);
    pizarra.stroke();
}

function dibujarAhorcado() {
    if (partesDibujadas < 7) {
        switch (partesDibujadas) {
            case 0:
                dibujarEstructura();
                break;
            case 1:
                dibujarCabeza();
                break;
            case 2:
                dibujarCuerpo();
                break;
        }
    }
}
```

```
    case 3:
        dibujarBrazoIzquierdo();
        break;
    case 4:
        dibujarBrazoDerecho();
        break;
    case 5:
        dibujarPiernaIzquierda();
        break;
    case 6:
        dibujarPiernaDerecha();
        break;
    }
    partesDibujadas++;
} else {
    reiniciarAhorcado();
}
}

function reiniciarAhorcado() {
    pizarra.clearRect(0, 0, canvas.width, canvas.height); // Limpiar el canvas
    partesDibujadas = 0; // Reiniciar partes dibujadas
}
```

## 4.2. Resultados



**Generar Ahorcado**

En la imagen se muestra el canvas con el 'el ahorcado' ya dibujado.

## 5. Repositorio

Repositorio GitHub