

```

function iniciarPrograma()
{
    //FECHA
    const fecha = new Date();
    let dia = fecha.getDate();
    let mes = fecha.getMonth() + 1;
    let anio = fecha.getFullYear();
    let fechaActual = `${dia}/${mes}/${anio}`;
    let inputFecha = document.getElementById('fecha');
    inputFecha.innerHTML = fechaActual;
    //Descarga de archivos
    document.getElementById('btnJs').addEventListener('click',function(){
        window.open('./jspdf.pdf','_blank');
    });
    document.getElementById('btnHTML').addEventListener('click',function(){
        window.open('./htmlpdf.pdf','_blank');
    });
    document.getElementById('btnCSS').addEventListener('click',function(){
        window.open('./csspdf.pdf','_blank');
    });
    //VISIBILIDAD DE SECCIONES
    document.getElementById('section-resolver-gj').style.display='none';
    document.getElementById('section-resolver-j').style.display='none';
    document.getElementById('section-resolver-gs').style.display='none';
    document.getElementById('section-resolver-d').style.display='none';
    document.getElementById('section-resolver-c').style.display='none';
    document.getElementById('section-resolver-p').style.display='none';
    document.getElementById('section-resolver-p-i').style.display='none';
    document.getElementById('portada').style.display='block';
    document.getElementById('introduccion').style.display='none';
    document.getElementById('menu-principal').style.display='none';
    document.getElementById('unidad-2-introduccion').style.display='none';
    document.getElementById('unidad-2-menu').style.display='none';
    document.getElementById('falsa-posicion-1').style.display='none';
    document.getElementById('falsa-posicion-2').style.display='none';
    document.getElementById('newton-1').style.display='none';
    document.getElementById('newton-2').style.display='none';
    document.getElementById('secante-1').style.display='none';
    document.getElementById('secante-2').style.display='none';
    document.getElementById('salida').style.display='none';
    document.getElementById('unidad-3-introduccion').style.display='none';
    document.getElementById('unidad-3-menu').style.display='none';
    document.getElementById('gauss-jordan-1').style.display='none';
    document.getElementById('gauss-jordan-2').style.display='none';
    document.getElementById('jacobi-1').style.display='none';
    document.getElementById('jacobi-2').style.display='none';
    document.getElementById('gauss-seidel-1').style.display='none';
    document.getElementById('gauss-seidel-2').style.display='none';
    document.getElementById('gauss-seidel-2').style.display='none';
    document.getElementById('unidad-4-introduccion').style.display='none';
    document.getElementById('unidad-4-menu').style.display='none';
    document.getElementById('doolittle-1').style.display='none';
    document.getElementById('doolittle-2').style.display='none';
    document.getElementById('cholesky-1').style.display='none';
    document.getElementById('cholesky-2').style.display='none';
    document.getElementById('unidad-5-introduccion').style.display='none';
    document.getElementById('unidad-5-menu').style.display='none';
    document.getElementById('potencia-1').style.display='none';
    document.getElementById('potencia-2').style.display='none';
}

```

```

document.getElementById('potencia-inversa-1').style.display='none';
document.getElementById('potencia-inversa-2').style.display='none';
document.getElementById('solucion-fp').style.display='none';
document.getElementById('solucion-n').style.display = 'none';
document.getElementById('solucion-s').style.display = 'none'
document.getElementById('resultado-gj').style.display = 'none '
//TRANSICIONES
botonSiguiente = document.getElementById('siguiente');
botonSiguiente.addEventListener('click',function()
{
    document.getElementById('portada').style.display='none';
    document.getElementById('introduccion').style.display='block'
});
botonRegresar = document.getElementById('regresar');
botonRegresar.addEventListener('click',function(){
    document.getElementById('portada').style.display='block';
    document.getElementById('introduccion').style.display='none'
});
botonSiguiente2 = document.getElementById('siguiente-2');
botonSiguiente2.addEventListener('click',function()
{
    document.getElementById('introduccion').style.display='none';
    document.getElementById('menu-principal').style.display='block'
});
botonRegresar2 = document.getElementById('regresar-2')
botonRegresar2.addEventListener('click',function(){
    document.getElementById('introduccion').style.display='block';
    document.getElementById('menu-principal').style.display='none';
})
//SALIDA
botonSalir = document.getElementById('salir')
botonSalir.addEventListener('click',salida);
botonRegresarU21 = document.getElementById('regresar-u-2-1');
botonRegresarU21.addEventListener('click',function(){
    document.getElementById('menu-principal').style.display='block'
    document.getElementById('unidad-2-introduccion').style.display='none'
});
botonSiguienteU21 = document.getElementById('siguiente-u-2-1');
botonSiguienteU21.addEventListener('click',function(){
    document.getElementById('unidad-2-introduccion').style.display='none'
    document.getElementById('unidad-2-menu').style.display='block'
})
let botonRegresarU22 = document.getElementById('regresar-u-2-2');
botonRegresarU22.addEventListener('click',function(){
    document.getElementById('unidad-2-menu').style.display='none'
    document.getElementById('unidad-2-introduccion').style.display='block'
})
//SELECCION DE UNIDAD
let botonSeleccionar = document.getElementById('seleccionar')
botonSeleccionar.addEventListener('click', seleccionarUnidad);
//SELECCION DE METODO UNIDAD II
let botonSiguienteU22 = document.getElementById('siguiente-u-2-2');
botonSiguienteU22.addEventListener('click',seleccionarU2);
//SELECCION DE METODO UNIDAD III
let botonSiguienteU32 = document.getElementById('siguiente-u-3-2');
botonSiguienteU32.addEventListener('click',seleccionarU3);
//SELECCION DE METODO UNIDAD IV
let botonSiguienteU42 = document.getElementById('siguiente-u-4-2');
botonSiguienteU42.addEventListener('click',seleccionarU4);

```

```

//SELECCION DE METODO UNIDAD V
let botonSiguienteU52 = document.getElementById('seleccionar-u-5-2');
botonSiguienteU52.addEventListener('click',seleccionarU5);
//RESOLUCION UNIDAD II
let botonResolverFP = document.getElementById('resolver-f-p');
botonResolverFP.addEventListener('click',resolverFP);
//
let botonResolverN = document.getElementById('resolver-n');
botonResolverN.addEventListener('click',resolverN);
//
let botonResolverS = document.getElementById('resolver-s');
botonResolverS.addEventListener('click',resolverS);
//RESOLUCION UNIDAD III
//Crear Matriz Gauss Jordan
let botonCrearMatriz = document.getElementById('crear-matriz');
botonCrearMatriz.addEventListener('click', function(){
    crearMatriz(1);
});
//Crear Matriz Jacobi
let botonCrearMatriz2 = document.getElementById('crear-matriz-2');
botonCrearMatriz2.addEventListener('click', function(){
    crearMatriz(2);
});
//Crear Matriz Gauss Seidel
let botonCrearMatriz3 = document.getElementById('crear-matriz-3');
botonCrearMatriz3.addEventListener('click', function(){
    crearMatriz(3);
});
//Resolver Gauss Jordan
botonResolverGJ = document.getElementById('resolver-gj');
botonResolverGJ.addEventListener('click',resolverGJ);
//Resolver Jacobi
botonResolverJ = document.getElementById('resolver-jacobi');
botonResolverJ.addEventListener('click',resolverJ);
//Resolver Gauss-Seidel
botonResolverGS = document.getElementById('resolver-gauss-seidel');
botonResolverGS.addEventListener('click',resolverGS);
//RESOLUCION UNIDAD IV
//Crear matriz Doolittle
let botonCrearMatriz4 = document.getElementById('crear-matriz-4');
botonCrearMatriz4.addEventListener('click', function(){
    crearMatriz(4);
});
//Resolver Doolittle
let botonResolverD = document.getElementById('resolver-doolittle');
botonResolverD.addEventListener('click',resolverD);
//Crear matriz Cholesky
let botonCrearMatriz5 = document.getElementById('crear-matriz-5');
botonCrearMatriz5.addEventListener('click', function(){
    crearMatriz(5);
});
//Resolver Cholesky
let botonResolverC = document.getElementById('resolver-cholesky');
botonResolverC.addEventListener('click',resolverC);
//RESOLUCION UNIDAD V
//Crear matriz Potencia
let botonCrearMatriz6 = document.getElementById('crear-matriz-6');
botonCrearMatriz6.addEventListener('click',function(){
    crearMatrizA(6);
});

```

```

        crearVectorInicial(6);
    });
    //Resolver Potencia
    let botonResolverP = document.getElementById('resolver-potencia');
    botonResolverP.addEventListener('click',resolverP);
    //Crear matriz Potencia Inversa
    let botonCrearMatriz7 = document.getElementById('crear-matriz-7');
    botonCrearMatriz7.addEventListener('click',function(){
        crearMatrizA(7);
        crearVectorInicial(7);
    });
    //Resolver Potencia Inversa
    let botonResolverPI = document.getElementById('resolver-potencia-inversa');
    botonResolverPI.addEventListener('click',resolverPI);
}
//SELECCION UNIDAD
function seleccionarUnidad()
{
    let inputUnidad2 = document.getElementById('u-2');
    let inputUnidad3 = document.getElementById('u-3');
    let inputUnidad4 = document.getElementById('u-4');
    let inputUnidad5 = document.getElementById('u-5');

    if(inputUnidad2.checked)
    {
        document.getElementById('menu-principal').style.display='none';
        document.getElementById('unidad-2-introduccion').style.display='block';
    }
    else if(inputUnidad3.checked)
    {
        document.getElementById('menu-principal').style.display='none';
        let a = document.getElementById('unidad-3-introduccion');
        a.style.display='block';
        let b = document.getElementById('unidad-3-menu');

        let botonRegresarU31 = document.getElementById('regresar-u-3-1')
        botonRegresarU31.addEventListener('click',function(){
            regresoMP(a);
        });
        let botonRegresarU32 = document.getElementById('regresar-u-3-2')
        botonRegresarU32.addEventListener('click',function(){
            regresoMI(a,b);
        });
        let botonSiguienteU31 = document.getElementById('siguiente-u-3-1')
        botonSiguienteU31.addEventListener('click',function(){
            siguienteU(a,b);
        })
    }
    else if(inputUnidad4.checked)
    {
        document.getElementById('menu-principal').style.display='none';

        let a = document.getElementById('unidad-4-introduccion');
        a.style.display='block';
        let b = document.getElementById('unidad-4-menu');

        let botonRegresoU41 = document.getElementById('regresar-u-4-1')
        botonRegresoU41.addEventListener('click',function(){

```

```

        regresoMP(a);
    });
    let botonRegresarU42 = document.getElementById('regresar-u-4-2')
    botonRegresarU42.addEventListener('click',function(){
        regresoMI(a,b);
    });
    let botonSiguienteU41 = document.getElementById('siguiente-u-4-1')
    botonSiguienteU41.addEventListener('click',function(){
        siguienteU(a,b);
    })
}
else if(inputUnidad5.checked)
{
    document.getElementById('menu-principal').style.display='none';
    let a = document.getElementById('unidad-5-introduccion');
    a.style.display='block';
    let b = document.getElementById('unidad-5-menu');

    let botonRegresoU31 = document.getElementById('regresar-u-5-1')
    botonRegresoU31.addEventListener('click',function(){
        regresoMP(a);
    });
    let botonRegresarU52 = document.getElementById('regresar-u-5-2')
    botonRegresarU52.addEventListener('click',function(){
        regresoMI(a,b);
    });
    let botonSiguienteU51 = document.getElementById('siguiente-u-5-1')
    botonSiguienteU51.addEventListener('click',function(){
        siguienteU(a,b);
    })
}
else
{
    alert('Ingresa una opción :');
}
}
//FUNCIONES PARA TRANSICIONES
function regresoMP(a){
    a.style.display = 'none'
    document.getElementById('menu-principal').style.display='block';
}
function regresoMI(a,b){
    a.style.display = 'block';
    b.style.display = 'none';
}
function siguienteU(a,b){
    a.style.display = 'none'
    b.style.display = 'block'
}
//SELECCION UNIDAD 2
function seleccionarU2()
{
    let inputFalsaPosicion = document.getElementById('falsa-posicion');
    let inputNewton = document.getElementById('newton');
    let inputSecante = document.getElementById('secante')

    if(inputFalsaPosicion.checked)
    {
        let a = document.getElementById('falsa-posicion-1')
    }
}

```

```

let b = document.getElementById('unidad-2-menu')
let c = document.getElementById('falsa-posicion-2')

a.style.display = 'block'
b.style.display = 'none'

let botonRegresarU2FP = document.getElementById('regresar-u-2-f-p');
botonRegresarU2FP.addEventListener('click',function(){
    a.style.display = 'none';
    b.style.display = 'block';
});
let botonRegresarU2FP2 = document.getElementById('regresar-u-2-f-p-2');
botonRegresarU2FP2.addEventListener('click',function(){
    a.style.display = 'block';
    c.style.display = 'none';
});
let botonSiguienteU2FP = document.getElementById('siguiente-u-2-f-p')
botonSiguienteU2FP.addEventListener('click',function(){
    a.style.display = 'none';
    c.style.display = 'block';
});
let botonSalir2 = document.getElementById('salir-2')
botonSalir2.addEventListener('click',salida)
}
else if(inputNewton.checked)
{
    let a = document.getElementById('newton-1')
    let b = document.getElementById('unidad-2-menu')
    let c = document.getElementById('newton-2')

    a.style.display = 'block';
    b.style.display = 'none';

    let botonRegresarU2N = document.getElementById('regresar-u-2-n');
    botonRegresarU2N.addEventListener('click',function(){
        a.style.display = 'none';
        b.style.display = 'block';
    })
    let botonRegresarU2N2 = document.getElementById('regresar-u-2-n-2');
    botonRegresarU2N2.addEventListener('click',function(){
        a.style.display = 'block';
        c.style.display = 'none';
    });
    let botonSiguienteU2N = document.getElementById('siguiente-u-2-n')
    botonSiguienteU2N.addEventListener('click',function(){
        a.style.display = 'none';
        c.style.display = 'block';
    });
    let botonSalir3 = document.getElementById('salir-3')
    botonSalir3.addEventListener('click',salida)
}
else if(inputSecante.checked)
{
    let a = document.getElementById('secante-1')
    let b = document.getElementById('unidad-2-menu')
    let c = document.getElementById('secante-2')

    a.style.display = 'block';
    b.style.display = 'none';

```

```

        let botonRegresarU2S = document.getElementById('regresar-u-2-s');
        botonRegresarU2S.addEventListener('click',function(){
            a.style.display = 'none';
            b.style.display = 'block';
        });
        let botonRegresarU2S2 = document.getElementById('regresar-u-2-s-2');
        botonRegresarU2S2.addEventListener('click',function(){
            a.style.display = 'block';
            c.style.display = 'none';
        });
        let botonSiguienteU2S = document.getElementById('siguiente-u-2-s');
        botonSiguienteU2S.addEventListener('click',function(){
            a.style.display = 'none';
            c.style.display = 'block';
        });
        let botonSalir4 = document.getElementById('salir-4');
        botonSalir4.addEventListener('click',salida)
    }
    else
    {
        alert('Ingresa una opcion :')
    }
}
//SELECCION UNIDAD 3
function seleccionarU3(){
    let inputGaussJordan = document.getElementById('gauss-jordan');
    let inputJacobi = document.getElementById('jacobi');
    let inputGaussSeidel = document.getElementById('gauss-seidel');

    if(inputGaussJordan.checked){
        let a = document.getElementById('gauss-jordan-1');
        let b = document.getElementById('unidad-3-menu');
        let c = document.getElementById('gauss-jordan-2');

        a.style.display = 'block';
        b.style.display = 'none';

        let botonRegresarGJ = document.getElementById('regresar-g-j');
        botonRegresarGJ.addEventListener('click',function(){
            a.style.display = 'none';
            b.style.display = 'block';
        });
        let botonRegresarGJ2 = document.getElementById('regresar-g-j-2');
        botonRegresarGJ2.addEventListener('click',function(){
            a.style.display = 'block';
            c.style.display = 'none';
        });
        let botonSiguienteGJ = document.getElementById('siguiente-g-j');
        botonSiguienteGJ.addEventListener('click',function(){
            a.style.display = 'none';
            c.style.display = 'block';
        });
        let botonSalir = document.getElementById('salir-5');
        botonSalir.addEventListener('click',salida);
    }
    else if(inputJacobi.checked){
        let a = document.getElementById('jacobi-1')
        let b = document.getElementById('unidad-3-menu')
    }
}

```

```

let c = document.getElementById('jacobi-2')

a.style.display = 'block'
b.style.display = 'none'

let botonRegresarJ = document.getElementById('regresar-j');
botonRegresarJ.addEventListener('click',function(){
    a.style.display = 'none';
    b.style.display = 'block';
});
let botonRegresarJ2 = document.getElementById('regresar-j-2');
botonRegresarJ2.addEventListener('click',function(){
    a.style.display = 'block';
    c.style.display = 'none';
});
let botonSiguienteJ = document.getElementById('siguiente-j')
botonSiguienteJ.addEventListener('click',function(){
    a.style.display = 'none';
    c.style.display = 'block';
});
let botonSalir = document.getElementById('salir-6');
botonSalir.addEventListener('click',salida);
}
else if(inputGaussSeidel.checked){
    let a = document.getElementById('gauss-seidel-1')
    let b = document.getElementById('unidad-3-menu')
    let c = document.getElementById('gauss-seidel-2')

    a.style.display = 'block'
    b.style.display = 'none'

    let botonRegresarGS = document.getElementById('regresar-g-s');
    botonRegresarGS.addEventListener('click',function(){
        a.style.display = 'none';
        b.style.display = 'block';
    });
    let botonRegresarGS2 = document.getElementById('regresar-g-s-2');
    botonRegresarGS2.addEventListener('click',function(){
        a.style.display = 'block';
        c.style.display = 'none';
    });
    let botonSiguienteGS = document.getElementById('siguiente-g-s')
    botonSiguienteGS.addEventListener('click',function(){
        a.style.display = 'none';
        c.style.display = 'block';
    });
    let botonSalir = document.getElementById('salir-7');
    botonSalir.addEventListener('click',salida);
}
else
{
    alert('Ingresa una opcion valida :');
}
}
//SELECCION UNIDAD 4
function seleccionarU4(){
    let inputdoolittle = document.getElementById('doolittle');
    let inputCholesky = document.getElementById('cholesky');

```



```

if(inputdoolittle.checked){
    let a = document.getElementById('doolittle-1');
    let b = document.getElementById('unidad-4-menu');
    let c = document.getElementById('doolittle-2');

    a.style.display = 'block'
    b.style.display = 'none'

    let botonRegresarC = document.getElementById('regresar-c');
    botonRegresarC.addEventListener('click',function(){
        a.style.display = 'none';
        b.style.display = 'block';
    });
    let botonRegresarC2 = document.getElementById('regresar-c-2');
    botonRegresarC2.addEventListener('click',function(){
        a.style.display = 'block';
        c.style.display = 'none';
    });
    let botonSiguienteC = document.getElementById('siguiente-c')
    botonSiguienteC.addEventListener('click',function(){
        a.style.display = 'none';
        c.style.display = 'block';
    });
    let botonSalir = document.getElementById('salir-8');
    botonSalir.addEventListener('click',salida);
}
else if(inputCholesky.checked){
    let a = document.getElementById('cholesky-1');
    let b = document.getElementById('unidad-4-menu');
    let c = document.getElementById('cholesky-2');

    a.style.display = 'block'
    b.style.display = 'none'

    let botonRegresarCsky = document.getElementById('regresar-csky');
    botonRegresarCsky.addEventListener('click',function(){
        a.style.display = 'none';
        b.style.display = 'block';
    });
    let botonRegresarCsky2 = document.getElementById('regresar-csky-2');
    botonRegresarCsky2.addEventListener('click',function(){
        a.style.display = 'block';
        c.style.display = 'none';
    });
    let botonSiguienteCsky = document.getElementById('siguiente-csky')
    botonSiguienteCsky.addEventListener('click',function(){
        a.style.display = 'none';
        c.style.display = 'block';
    });
    let botonSalir = document.getElementById('salir-9');
    botonSalir.addEventListener('click',salida);
}
else{
    alert('Ingresa una opción válida :');
}
}
//SELECCION UNIDAD V
function seleccionarU5(){
    let inputPotencia = document.getElementById('potencia');

```

```

let inputPotenciaInversa = document.getElementById('potencia-inversa');

if(inputPotencia.checked){
    let a = document.getElementById('potencia-1');
    let b = document.getElementById('unidad-5-menu');
    let c = document.getElementById('potencia-2');

    a.style.display = 'block';
    b.style.display = 'none';

    let botonRegresarP = document.getElementById('regresar-p');
    botonRegresarP.addEventListener('click', function(){
        a.style.display = 'none';
        b.style.display = 'block';
    });
    let botonRegresarP2 = document.getElementById('regresar-p-2');
    botonRegresarP2.addEventListener('click', function(){
        a.style.display = 'block';
        c.style.display = 'none';
    });
    let botonSiguienteP = document.getElementById('siguiente-p');
    botonSiguienteP.addEventListener('click', function(){
        a.style.display = 'none';
        c.style.display = 'block';
    });
    let botonSalir = document.getElementById('salir-9');
    botonSalir.addEventListener('click', salida);
}
else if(inputPotenciaInversa.checked){
    let a = document.getElementById('potencia-inversa-1');
    let b = document.getElementById('unidad-5-menu');
    let c = document.getElementById('potencia-inversa-2');

    a.style.display = 'block';
    b.style.display = 'none';

    let botonRegresarPi = document.getElementById('regresar-p-i');
    botonRegresarPi.addEventListener('click', function(){
        a.style.display = 'none';
        b.style.display = 'block';
    });
    let botonRegresarPi2 = document.getElementById('regresar-p-i-2');
    botonRegresarPi2.addEventListener('click', function(){
        a.style.display = 'block';
        c.style.display = 'none';
    });
    let botonSiguientePi = document.getElementById('siguiente-p-i');
    botonSiguientePi.addEventListener('click', function(){
        a.style.display = 'none';
        c.style.display = 'block';
    });
    let botonSalir = document.getElementById('salir-10');
    botonSalir.addEventListener('click', salida);
}
else{
    alert('Ingresa una opción válida :');
}
}
//RESOLVER FALSA POSICION

```

```

function resolverFP(){
    let a = parseFloat(document.getElementById('inter-1').value);
    let b = parseFloat(document.getElementById('inter-2').value);
    let inputEcuacion1 = document.getElementById('ecuacion-1-f-p');
    let inputEcuacion2 = document.getElementById('ecuacion-2-f-p');
    let tablaFP = document.getElementById('tabla-f-p');
    let n;
    let cen = true;

    if(inputEcuacion1.checked){
        let table = '<h1>Tabla Falsa Posición</h1><table border="1" class="tabla-
metodos">';
        table += '<thead><tr><th width="50px">n</th><th width="50px">a</th><th
width="50px">b</th><th width="50px">c</th><th width="50px">fa</th><th
width="50px">fb</th><th width="50px">fc</th></tr></thead>'; // Encabezados de la
tabla
        table += '<tbody>';

        let c;
        let fc;
        n = 1;

        do {
            table += '<tr>';

            let fa = a * Math.exp(a) - a**3 - 3;
            let fb = b * Math.exp(b) - b**3 - 3;
            c = a - fa * ((a - b) / (fa - fb));
            fc = c * Math.exp(c) - c**3 - 3;
            if (fa * fc > 0) {
                a = c; // Si fa y fc tienen el mismo signo, actualiza 'a'
            } else {
                b = c; // Si fa y fc tienen signos opuestos, actualiza 'b'
            }
            // Agregar los valores a la tabla
            table += '<td>${n}</td>';
            table += '<td>${a.toFixed(4)}</td>';
            table += '<td>${b.toFixed(4)}</td>';
            table += '<td>${c.toFixed(4)}</td>';
            table += '<td>${fa.toFixed(4)}</td>';
            table += '<td>${fb.toFixed(4)}</td>';
            table += '<td>${fc.toFixed(4)}</td>';
            table += '</tr>';
            n++;
            if(n>100)
            {
                break;
            }

        } while (Math.abs(fc) > 0.0001);
        if(n>100)
        {
            alert('El intervalo no es adecuado');
            cen = false;
        }
        if(cen){
            table += '</tbody></table>';
            tablaFP.innerHTML = table;
            document.getElementById('sol-f-p').innerHTML = c.toFixed(4);
        }
    }
}

```

```

        document.getElementById('solucion-fp').style.display = 'block'
    }
}
else if(inputEcuacion2.checked)
{
    let table = '<h1>Tabla Falsa Posición</h1><table border="1" class="tabla-
metodos">';
    table += '<thead><tr><th width="50px">n</th><th width="50px">a</th><th
width="50px">b</th><th width="50px">c</th><th width="50px">fa</th><th
width="50px">fb</th><th width="50px">fc</th></tr></thead>';
    table += '<tbody>';
    let c;
    let fc;
    let n=1;
    do
    {
        table += '<tr>'
        let fa = a**2 - Math.exp(a) - 3 * a + 2;
        let fb = b**2 - Math.exp(b) - 3 * b + 2;
        c = a - fa * ((a - b) / (fa - fb)); //
        fc = c**2 - Math.exp(c) - 3 * c + 2; //
        if (fa * fc > 0) {
            a = c; // Si fa y fc tienen el mismo signo, actualiza 'a'
        }
        if (fb * fc > 0) {
            b = c; // Si fb y fc tienen el mismo signo, actualiza 'b'
        }
        // Agregar los valores a la tabla
        table += '<td>${n}</td>`
        table += '<td>${a.toFixed(4)}</td>`;
        table += '<td>${b.toFixed(4)}</td>`;
        table += '<td>${c.toFixed(4)}</td>`;
        table += '<td>${fa.toFixed(4)}</td>`;
        table += '<td>${fb.toFixed(4)}</td>`;
        table += '<td>${fc.toFixed(4)}</td>`;
        table += '</tr>';
        n++;
    } while (Math.abs(fc)>=0.0001);

    table += '</tbody></table>';
    tablaFP.innerHTML = table;
    document.getElementById('sol-f-p').innerHTML = c.toFixed(4);

    document.getElementById('solucion-fp').style.display='block';
}
else
{
    alert('Selecciona una ecuación para resolver');
}
}
//RESOLVER NEWTON
function resolverN(){
    let x = parseFloat(document.getElementById('valor-inicial').value);
    let tablaN = document.getElementById('tabla-n');
    let fx;
    let n=1;

    let inputEcuacion1 = document.getElementById('ecuacion-1-n');
    let inputEcuacion2 = document.getElementById('ecuacion-2-n');

```

```

        if(inputEcuacion1.checked){
            let table = '<h1>Tabla Newton</h1><table border="1" class="tabla-
metodos">'; // Agregar un borde a la tabla
            table += '<thead><tr> <th width="50px"> \\(n\\) </th> <th
width="50px">\\(x_n\\) </th> <th width="50px">\\(f(x)\\) </th> <th
width="50px">\\(f'(x)\\) </th></tr></thead>'; // Encabezados de la tabla
            table += '<tbody>'; // Cuerpo de la tabla
            do{
                fx = x*Math.sin(x) + 2*x**2-1;
                let fdx = x*Math.cos(x)+Math.sin(x)+4*x;

                table += '<td>${n}</td>`
                table += '<td>${x.toFixed(4)}</td>`
                table += '<td>${fx.toFixed(4)}</td>`
                table += '<td>${fdx.toFixed(4)}</td>`
                table += '</tr>';

                x = x - (fx/fdx);
                n++;
            }while(Math.abs(fx)>0.00001);
            tablaN.innerHTML=table;
            document.getElementById('sol-n').innerHTML = x.toFixed(4);
            document.getElementById('solucion-n').style.display = 'block'
            MathJax.typeset();
        }
        else if(inputEcuacion2.checked){
            let table = '<h1>Tabla Newton</h1><table border="1" class="tabla-
metodos">';
            table += '<thead><tr> <th width="50px">\\(n\\) </th> <th
width="50px">\\(x_n\\) </th> <th width="50px">\\(f(x)\\) </th> <th
width="50px">\\(f'(x)\\) </th></tr></thead>'; // Encabezados de la tabla
            table += '<tbody>'; // Cuerpo de la tabla

            let n = 1; // Inicializar n
            let fx;

            do {
                fx = x**2 - Math.exp(x) - 3 * x + 2; // Cálculo de f(x)
                let fdx = 2 * x - Math.exp(x) - 3; // Cálculo de f'(x)

                // Agregar los valores a la tabla
                table += '<tr>`;
                table += '<td>${n}</td>`; // Iteración
                table += '<td>${x.toFixed(4)}</td>`; // Valor de x
                table += '<td>${fx.toFixed(4)}</td>`; // Valor de f(x)
                table += '<td>${fdx.toFixed(4)}</td>`; // Valor de f'(x)
                table += '</tr>`;

                x = x - (fx / fdx); // Actualización de x con el método de Newton-
Raphson
                n++; // Incrementar el contador

            } while (Math.abs(fx) > 0.00001); // Salir cuando f(x) converja a 4
            decimales

            table += '</tbody></table>'; // Cerrar las etiquetas tbody y table

            // Mostrar la tabla en el elemento con id 'tablaN'

```

```

        tablaN.innerHTML = table;

        // Mostrar la solución final en el elemento con id 'sol-n'
        document.getElementById('sol-n').innerHTML = x.toFixed(4);

        // Mostrar la sección donde está la solución
        document.getElementById('solucion-n').style.display = 'block';
        MathJax.typeset();
    }
    else{
        alert('Ingresa una solución para resolver');
    }
}
//RESOLVER SECANTE
function resolverS(){
    let x = parseFloat(document.getElementById('valor-inicial-2').value);
    let x0 = 0;
    let fx;
    let fx0;
    let tablaS = document.getElementById('tabla-s');
    let n = 1;

    let inputEcuacion1 = document.getElementById('ecuacion-1-s');
    let inputEcuacion2 = document.getElementById('ecuacion-2-s');

    if(inputEcuacion1.checked){
        let table = '<h1>Tabla Secante</h1><br><table border="1" class="tabla-
metodos">'; // Agregar un borde a la tabla
        table += '<thead><tr> <th width="80px">n</th> <th
width="80px">\\(x_n\\)</th> <th width="80px">\\(f(x)\\)</th></tr></thead>'; //
Encabezados de la tabla
        table += '<tbody>'; // Cuerpo de la tabla
        fx0 = x0*Math.sin(x0) + 2*x0**2-1;
        table += '<td>${n}</td>'
        table += '<td>${x0.toFixed(4)}</td>'
        table += '<td>${fx0.toFixed(4)}</td>'
        table += '</tr>';
        do{
            n++;
            fx0 = x0*Math.sin(x0) + 2*x0**2-1;
            fx = x*Math.sin(x) + 2*x**2-1;

            table += '<td>${n}</td>'
            table += '<td>${x.toFixed(4)}</td>'
            table += '<td>${fx.toFixed(4)}</td>'
            table += '</tr>';

            let auxX = x - ((x-x0)/(fx-fx0))*fx;
            x0 = x;
            x = auxX;
        }while(Math.abs(fx)>0.00001||n>100);
        table += '</tbody></table>';
        tablaS.innerHTML=table;
        document.getElementById('sol-s').innerHTML = x.toFixed(4);
        document.getElementById('solucion-s').style.display = 'block';
        MathJax.typeset();
    }
    else if(inputEcuacion2.checked){

```

```

        let table = '<h1>Tabla Secante</h1><table border="1" class="tabla-
metodos">';
        table += '<thead><tr> <th width="80px">\\(n\\)</th> <th
width="80px">\\(x_n\\)</th> <th width="80px">\\(f(x)\\)</th></tr></thead>';
        table += '<tbody>';

        fx0 = x0 ** 2 - Math.exp(x0) - 3 * x0 + 2;
        table += '<tr><td>${n}</td>`;
        table += '<td>${x0.toFixed(4)}</td>`;
        table += '<td>${fx0.toFixed(4)}</td></tr>`;

        do {
            n++;
            fx0 = x0 ** 2 - Math.exp(x0) - 3 * x0 + 2;
            fx = x ** 2 - Math.exp(x) - 3 * x + 2;

            table += '<tr><td>${n}</td>`;
            table += '<td>${x.toFixed(4)}</td>`;
            table += '<td>${fx.toFixed(4)}</td></tr>`;

            let auxX = x - ((x - x0) / (fx - fx0)) * fx;
            x0 = x;
            x = auxX;

        } while (Math.abs(fx) > 0.00001 && n <= 100);

        table += '</tbody></table>';
        tablaS.innerHTML = table;
        document.getElementById('sol-s').innerHTML = x.toFixed(4);
        document.getElementById('solucion-s').style.display = 'block';
        MathJax.typeset();
    }
    else{
        alert('Ingresa una ecuación a resolver :');
    }
}
//CREAR MATRIZ
function crearMatriz(ord){
    document.getElementById('section-resolver-gj').style.display='block';
    document.getElementById('section-resolver-j').style.display='block';
    document.getElementById('section-resolver-gs').style.display='block';
    document.getElementById('section-resolver-d').style.display='block';
    document.getElementById('section-resolver-c').style.display='block';
    let n = parseInt(document.getElementById(`orden-${ord}`).value);
    let matriz;
    let ind;
    if(ord==1){
        matriz = document.getElementById('matriz-gj');
        ind = 'gj';
    }
    else if(ord==2){
        matriz = document.getElementById('matriz-j');
        ind = 'j';
    }
    else if(ord==3){
        matriz = document.getElementById('matriz-gs');
        ind = 'gs';
    }
    else if(ord==4){

```

```

        matriz = document.getElementById('matriz-d');
        ind = 'd';
    }
    else if(ord==5){
        matriz = document.getElementById('matriz-c');
        ind = 'c';
    }
    //Creación de la tabla que sobrescribirá el div en el doc
    let table = '';
    table += '<table border="0">';
    let cen=true;
    if(n<1||n>10){
        cen=false;
        alert('El valor de la matriz es inválido')
    }
    if(cen){
        for (let i = 0; i < n; i++) {
            table += '<tr>';
            for (let j = 0; j < n+1; j++) {
                let sg = "";
                let vr='';
                if(j==n){
                    sg = '=';
                    vr = ` `;
                }
                else{
                    sg='+'
                    vr=`\\(x_${j+1}\\)`;
                }
                if(j==0){
                    sg='';
                }
                table += `<td>${sg}<label for="matriz-${i}-${j}-${ind}"></label>
<input type="text" id="matriz-${i}-${j}-${ind}" placeholder="a${i+1},${j+1}"
class="inter">${vr}</td>`;
            }
            table += '</tr>';
        }
    }
    table += '</table>';
    //Si el centinela resulta falso, no se mostrará el boton resolver
    if(cen==false){
        document.getElementById('section-resolver-gj').style.display='none';
        document.getElementById('section-resolver-j').style.display='none';
        document.getElementById('section-resolver-gs').style.display='none';
        document.getElementById('section-resolver-d').style.display='none';
        document.getElementById('section-resolver-c').style.display='none';
    }
    matriz.innerHTML = table;
    MathJax.typeset(); //Forzar renderización con la API
}

function crearMatrizA(ord){
    let n = parseInt(document.getElementById(`orden-${ord}`).value);
    let matriz;
    let ind;
    if(ord==6){
        matriz = document.getElementById('matriz-p');
        ind = 'p';
    }
}

```



```

else if(ord==7){
    matriz = document.getElementById('matriz-p-i');
    ind = 'p-i';
}
let cen=true;
if(n<1||n>10){
    cen=false;
    alert('El valor de la matriz es inválido')
}
let table = `\\(A=\\)<table>`;
if (cen) {
    document.getElementById('section-resolver-p').style.display='block';
    document.getElementById('section-resolver-p-i').style.display='block';
    for (let i = 0; i < n; i++) {
        table += '<tr>';
        for (let j = 0; j < n; j++) {
            table += `<td><label for="matriz-${i}-${j}-${ind}"></label><input
id="matriz-${i}-${j}-${ind}" class="inter" type="text"></input></td>`;
        }
        table += '</tr>';
    }
    table += '</table>';
    matriz.innerHTML = table;
    MathJax.typesetPromise([matriz]);
}
}
//RESOLVER GJ
function resolverGJ(){
    let n = parseInt(document.getElementById('orden-1').value);
    var matriz = [];
    let cen = true;
    for (let i = 0; i < n && cen; i++) {
        matriz[i] = [];
        for (let j = 0; j < n + 1; j++) {
            let valor = document.getElementById(`matriz-${i}-${j}-gj`).value;
            if (valor === "") {
                alert('El valor de los coeficientes no puede estar vacío');
                cen = false;
                break;
            }
            matriz[i][j] = parseInt(valor);
        }
    }
    for (let i = 0; i < n; i++) {
        let pivote = matriz[i][i];
        if (pivote === 0) {
            alert("El elemento pivote es cero, no se puede emplear el método");
            return;
        }
        for (let j = 0; j < n + 1; j++) {
            matriz[i][j] /= pivote;
        }
        for (let k = 0; k < n; k++) {
            if (k !== i) {
                let factor = matriz[k][i];
                for (let j = 0; j < n + 1; j++) {
                    matriz[k][j] -= factor * matriz[i][j];
                }
            }
        }
    }
}

```

```

    }
}
//Obtener los resultados de la ultima columna de la matriz aumentada
let solucion = document.getElementById('r-gj');
let resultados = "";
for(i=0;i<n;i++){
    resultados += `\\(x_{{i+1}} = ${matriz[i][n].toFixed(4)}\\) <br>`;
}
solucion.innerHTML = resultados;
MathJax.typeset();
document.getElementById('resultado-gj').style.display = 'block'
}
//RESOLVER JACOBI
function resolverJ(){
    let n = parseInt(document.getElementById('orden-2').value);
    var matriz = [];
    let cen = true;
    // Leer la matriz desde el documento
    for (let i = 0; i < n && cen; i++) {
        matriz[i] = [];
        for (let j = 0; j < n + 1; j++) {
            let valor = document.getElementById(`matriz-${i}-${j}-j`).value;
            // Verificar si el valor está vacío o nulo
            if (valor === "") {
                alert('El valor de los coeficientes no puede estar vacío');
                cen = false;
                break;
            }
            matriz[i][j] = parseInt(valor);
        }
    }
    //Verificar si la matriz de la diagonal es dominante
    cen=true;
    for (let i = 0; i < n && cen; i++) {
        for (let j = 0; j < n-1 + 1; j++) {
            if(i!=j){
                if (matriz[i][i]<matriz[i][j]) {
                    alert('La diagonal no es dominante');
                    cen = false;
                    break;
                }
            }
        }
    }
}
if(cen){
    let sol = Array(n).fill(0);
    let auxSol = Array(n).fill(0);
    let convergencia;
    let table = `<h2>Tabla Jacobi</h2><br><table class="tabla-metodos"
border="1"><thead><th>n</th>`;
    let tablaJacobi = document.getElementById('tabla-jacobi');
    let iter=0;
    //Encabezados
    for(i=0;i<n;i++){
        table+=`<th width="50px">\\( \\mathbf{x_{{i+1}}} \\)</th>`;
    }
    table+=`</thead>`;
    do {
        convergencia = true;

```

```

        for (let i = 0; i < n; i++) {
            let x = matriz[i][i] ** -1 * matriz[i][n];
            for (let j = 0; j < n; j++) {
                if (i !== j) {
                    x += matriz[i][j] ** -1 * (-matriz[i][j] * sol[j]);
                }
            }
            auxSol[i] = x;
        }
        //Creacion de tabla de iteraciones
        iter++;
        table+=`<tr><td width="50px" align="center">${iter}</td>`;
        for(i=0;i<n;i++){
            table += `<td width="70px"
align="center">${(auxSol[i]).toFixed(4)}</td>`;
        }
        table+=`</tr>`;
        // Verifica convergencia con el último valor conocido de cada
        variable, asegurando que la osilacion se minima
        for (let i = 0; i < n; i++) {
            if (Math.abs(sol[i] - auxSol[i]) > 0.0001) {
                convergencia = false;
                break;
            }
        }
        // Actualizamos sol para la próxima iteración
        sol = [...auxSol];
    } while (!convergencia);
    table+="  
</table>"
    let resultadosJacobi = document.getElementById('r-j');
    let resultados="Los Resultados son: <br>";
    for (let i = 0; i < n; i++) {
        resultados += `\\(x_{i+1}) \\thickapprox
${(sol[i]).toFixed(4)}\\)<br>`;
    }
    resultados += "<br><br>";
    tablaJacobi.innerHTML = table;
    resultadosJacobi.innerHTML = resultados;
    MathJax.typeset();
}

}

//Resolver Gauss-Seidel
function resolverGS(){
    let n = parseInt(document.getElementById('orden-3').value);
    var matriz = [];
    let cen = true; // Centinela
    // Leer la matriz desde el documento
    for (let i = 0; i < n && cen; i++) {
        matriz[i] = [];
        for (let j = 0; j < n + 1; j++) {
            let valor = document.getElementById(`matriz-${i}-${j}-gs`).value;
            // Verificar si el valor está vacío o nulo
            if (valor === "") {
                alert('El valor de los coeficientes no puede estar vacío');
                cen = false; // Marcar la matriz como no válida
                break; // Salir del bucle interior
            }
        }
        // Convertir a número e insertar en la matriz
        matriz[i][j] = parseInt(valor);
    }
}

```

```

    }
}
//Verificar si la matriz de la diagonal es estrictamente dominante
cen=true;
let sum=0;
let vdiag=0;
for (let i = 0; i < n && cen; i++) {
    for (let j = 0; j < n-1 + 1; j++) {
        vdiag = matriz[i][i];
        if(i!=j){
            sum += matriz[i][j];
            if(vdiag<sum){
                alert('La matriz no es estrictamente dominante');
                break;
                cen=false;
            }
        }
    }
}
}
if(cen){
    let sol = Array(n).fill(0);
    let convergencia;
    let table = `

## 


```

```

        for (let i = 0; i < n; i++) {
            resultados += `\\(x_{i+1} \\thickapprox
${(sol[i]).toFixed(4)}\\)<br>`;
        }
        resultados += "<br><br>";
        //Sobreescritura de la tabla y los resultados
        resultadosGaussSeidel.innerHTML=resultados;
        tablaGaussSeidel.innerHTML = table;
        //Forzar renderización
        MathJax.typeset();
    }
}
//UNIDAD IV
//Resolver Doolittle
function resolverD(){
    //Leer Matriz A del documento
    let n = parseInt(document.getElementById('orden-4').value);
    let cen = true;
    let matriz = [];
    for(let i=0;i<n;i++){
        matriz[i]=[];
        for(let j=0;j<n;j++){
            let value = parseInt(document.getElementById(`matriz-${i}-${j}-
d`).value);
            if(value==null){
                cen = false;
                alert('El valor de los coeficientes no puede estar vacío');
                break;
            }
            else{
                matriz[i][j] = value;
            }
        }
    }
}
//Factorización LU Doolittle l_kk = 1
let matrizL = [];
let matrizU = [];
for (let i = 0; i < n; i++) {
    matrizL[i] = [];
    matrizU[i] = []
    for (let j = 0; j < n; j++) {
        if(i==j){
            matrizL[i][j] = 1;
        }
        else{
            matrizL[i][j] = 0;
        }
        matrizU[i][j] = 0;
    }
}
for(let k=0;k<n;k++){
    let sum=0;
    for(let s=0;s<k;s++){
        sum += matrizL[k][s]*matrizU[s][k];
    }
    matrizU[k][k] = matriz[k][k]-sum;
    for(j=k+1;j<n;j++){
        let sum2 = 0;
        for(let s=0;s<k;s++){

```

```

        sum2 += matrizL[k][s] * matrizU[s][j];
    }
    matrizU[k][j] = (matriz[k][j]-sum2)/matrizL[k][k];
}
for(i=k+1;i<n;i++){
    let sum3 = 0;
    for(s=0;s<k;s++){
        sum3 += matrizL[i][s] * matrizU[s][k];
    }
    matrizL[i][k] = (matriz[i][k]-sum3)/matrizU[k][k];
}
}
//Calculo del vector solución
//Lz=b
let matrizLzb = [];
let z = [];
for (let i = 0; i < n; i++) {
    matrizLzb[i] = [];
    for (let j = 0; j < n+1; j++) {
        if(j!=n){
            matrizLzb[i][j] = matrizL[i][j];
        }
        else{
            matrizLzb[i][j] = parseInt(document.getElementById(`matriz-${i}-${n}-d`).value);
        }
    }
}
//Sustitución Progresiva
let suma=0;
z[0]=matrizLzb[0][n]/matrizLzb[0][0];
for(i=0;i<n;i++){
    suma=0;
    let b = matrizLzb[i][n];
    for (let j = 0; j < i; j++) {
        suma += matrizLzb[i][j]*z[j];
    }
    z[i] = matrizLzb[i][i] ** -1 * (b-suma);
}
//Ux=z
let matrizUxz = [];
for (let i = 0; i < n; i++) {
    matrizUxz[i] = [];
    for (let j = 0; j < n+1; j++) {
        if(j!=n){
            matrizUxz[i][j] = matrizU[i][j];
        }
        else{
            matrizUxz[i][j] = parseInt(document.getElementById(`matriz-${i}-${n}-d`).value);
        }
    }
}
//Sustitución Regresiva
let x = [];
for (let i = n-1; i >= 0; i--) {
    let b = matrizUxz[i][n];
    let suma = 0;
    for (let j = n-1; j > i; j--) {

```

```

        suma += matrizUxz[i][j] * x[j];
    }
    x[i] = matrizUxz[i][i] ** -1 * (b-suma);
}
//Creación de matrices con formato LaTeX
let tableL = `\\( L = \\begin{pmatrix}`;
let tableU = `\\( U = \\begin{pmatrix}`;
let vectorZ = `\\( \\vec{z} = \\begin{pmatrix}`;
let vectorX = `\\( \\vec{x} = \\begin{pmatrix}`;
for(i=0;i<n;i++){
    for(j=0;j<n;j++){
        tableL += `${matrizL[i][j]} `;
        tableU += `${matrizU[i][j]} `;
        if(j!=n-1){
            tableL += '&';
            tableU += '&';
        }
    }
    vectorZ += `${z[i]}\\\\`;
    vectorX += `${x[i]}\\\\`;
    tableL += '\\\\';
    tableU += '\\\\';
}
tableL += ' \\end{pmatrix} \\\';
tableU += ' \\end{pmatrix} \\\';
vectorZ += ' \\end{pmatrix} \\\';
vectorX += ' \\end{pmatrix} \\\';

//Sobreescritura en el documento
let matrizLDoolittle = document.getElementById('matriz-l-d');
let matrizUDoolittle = document.getElementById('matriz-u-d');
let vectorZD = document.getElementById('vector-z-d');
let resultadosDoolittle = document.getElementById('r-d');
//Llamar 'oso' a las variables importantes ;)
matrizLDoolittle.innerHTML = tableL;
matrizUDoolittle.innerHTML = tableU;
vectorZD.innerHTML = vectorZ;
resultadosDoolittle.innerHTML = vectorX;
MathJax.typeset();
}
//Resolver Cholesky
function resolverC(){
    //Leer Matriz A del documento
    let n = parseInt(document.getElementById('orden-5').value);
    let cen = true;
    let matriz = [];
    for(let i=0;i<n;i++){
        matriz[i]=[];
        for(let j=0;j<n;j++){
            let value = parseInt(document.getElementById(`matriz-${i}-${j}-c` ).value);
            if(value==null){
                cen = false;
                alert('El valor de los coeficientes no puede estar vacío');
                break;
            }
            else{
                matriz[i][j] = value;
            }
        }
    }
}

```

```

    }
}
//Verificar condiciones
//Simetrica
for (let i = 0; i < n; i++) {
    for (let j = 0; j < n; j++) {
        if(i!=j){
            if (matriz[i][j]!=matriz[j][i]) {
                cen = false;
                alert("La matriz no es simétrica");
                break;
            }
            if(!cen){
                break;
            }
        }
    }
}
//Definida positiva
// Verificar el criterio de Sylvester
for (let k = 1; k <= n; k++) {
    const subMatriz = matriz.slice(0, k).map(row => row.slice(0, k));
    if (determinante(subMatriz) <= 0) {
        cen = false;
        alert('La matriz no es definida positiva');
    }
}
if(cen){
    //Inicializar L y U
    let L = [];
    let U = [];
    for (let i = 0; i < n; i++) {
        L[i] = [];
        U[i] = [];
        for (let j = 0; j < n; j++) {
            L[i][j] = 0;
            U[i][j] = 0;
        }
    }
    //Algoritmo Cholesky
    let sum1 = 0;
    let sum2 = 0;
    for (let k = 0; k < n; k++) {
        sum1 = 0;
        sum2 = 0;
        for (let s = 0; s < k; s++) {
            sum1 += L[k][s] ** 2;
        }
        L[k][k] = Math.sqrt(matriz[k][k] - sum1);
        for (let i = k+1; i < n; i++) {
            for (let s = 0; s < k; s++) {
                sum2 += L[i][s]*L[k][s];
            }
            L[i][k] = (matriz[i][k]-sum2)/L[k][k];
        }
    }
    //Calcular U
    for (let i = 0; i < n; i++) {
        for (let j = 0; j < n; j++) {

```



```

        U[i][j] = L[j][i];
    }
}
//Calculo del vector solución
//Lz=b
let matrizLzb = [];
let z = [];
for (let i = 0; i < n; i++) {
    matrizLzb[i] = [];
    for (let j = 0; j < n+1; j++) {
        if(j!=n){
            matrizLzb[i][j] = L[i][j];
        }
        else{
            matrizLzb[i][j] = parseInt(document.getElementById(`matriz-
${i}-${n}-c`).value);
        }
    }
}
//Sustitución Progresiva
let suma=0;
z[0]=matrizLzb[0][n]/matrizLzb[0][0];
for(i=0;i<n;i++){
    suma=0;
    let b = matrizLzb[i][n];
    for (let j = 0; j < i; j++) {
        suma += matrizLzb[i][j]*z[j];
    }
    z[i] = matrizLzb[i][i] ** -1 * (b-suma);
}
//Ux=z
let matrizUxz = [];
for (let i = 0; i < n; i++) {
    matrizUxz [i] = [];
    for (let j = 0; j < n+1; j++) {
        if(j!=n){
            matrizUxz[i][j] = U[i][j];
        }
        else{
            matrizUxz[i][j] = z[i];
        }
    }
}
//Sustitución Regresiva
let x = [];
for (let i = n-1; i >= 0; i--) {
    let b = matrizUxz[i][n];
    let suma = 0;
    for (let j = n-1; j > i; j--) {
        suma += matrizUxz[i][j] * x[j];
    }
    x[i] = matrizUxz[i][i] ** -1 * (b-suma);
}
//Creación de L U y vectores con formato LaTeX
let tableL = `\\(L = \\begin{pmatrix}`;
let tableU = `\\(U = \\begin{pmatrix}`;
let vectorZ = `\\( \\vec{z} = \\begin{pmatrix}`;
let vectorX = `\\( \\vec{x} = \\begin{pmatrix}`;
for (let i = 0; i < n; i++) {

```

```

        for (let j = 0; j < n; j++) {
            tableL += `${(L[i][j].toFixed(4))}`;
            tableU += `${(U[i][j].toFixed(4))}`;
            if(j!=n-1){
                tableL += '&';
                tableU += '&';
            }
        }
        vectorZ += `${(z[i].toFixed(4))}\\\\`;
        vectorX += `${(x[i].toFixed(4))}\\\\`;
        tableL += `\\\\`;
        tableU += `\\\\`;
    }
    tableL += `\\end{pmatrix}\\`;
    tableU += `\\end{pmatrix}\\`;
    vectorX += `\\end{pmatrix}\\`;
    vectorZ += `\\end{pmatrix}\\`;
    //Sobreescritura en el documento
    let matrizLC = document.getElementById('matriz-l-c');
    let matrizUC = document.getElementById('matriz-u-c');
    let vectorZC = document.getElementById('vector-z-c');
    let vectorXC = document.getElementById('r-c');

    matrizLC.innerHTML = tableL;
    matrizUC.innerHTML = tableU;
    vectorZC.innerHTML = vectorZ;
    vectorXC.innerHTML = vectorX;
    MathJax.typeset();
}
}
//Determinante de la matriz
function determinante(matriz) {
    const n = matriz.length;
    if (n === 1) return matriz[0][0];
    if (n === 2) return matriz[0][0] * matriz[1][1] - matriz[0][1] *
matriz[1][0];

    let det = 0;
    for (let j = 0; j < n; j++) {
        const subMatriz = matriz.slice(1).map(row => row.filter((_, colIndex) =>
colIndex !== j));
        det += ((j % 2 === 0 ? 1 : -1) * matriz[0][j] * determinante(subMatriz));
    }
    return det;
}
//Crear vector Inicial
function crearVectorInicial(tam){
    let n = parseInt(document.getElementById(`orden-${tam}`).value);
    let ind;
    let vector;
    if(tam==6){
        vector = document.getElementById('vector-p');
        ind = 'p';
    }
    else if(tam==7) {
        vector = document.getElementById('vector-p-i');
        ind = 'p-i'
    }
    let table = `\\(x=\\)<table><tr><td>`;

```

```

        for (let i = 0; i < n; i++) {
            table += `<label for="vector-${i}-${ind}"></label><input id="vector-${i}-${ind}" class="inter"></input><br>`;
        }
        table += `</td></tr></table>`;
        vector.innerHTML = table;
        MathJax.typeset();
    }
    //Resolver Potencia
    function resolverP(){
        //Leer Matriz A del documento
        let n = parseInt(document.getElementById('orden-6').value);
        let cen = true;
        let matriz = [];
        let x = [];
        for(let i=0;i<n;i++){
            matriz[i]=[];
            let valorX = parseFloat(document.getElementById(`vector-${i}-p`).value);
            if(valorX == null){
                alert('El valor de los coeficientes no puede estar vacio');
                cen = false;
                break;
            }
            else x.push(valorX);
            for(let j=0;j<n;j++){
                let value = parseFloat(document.getElementById(`matriz-${i}-${j}-p`).value);
                if(value == null){
                    cen = false;
                    alert('El valor de los coeficientes no puede estar vacío');
                    break;
                }
                else{
                    matriz[i][j] = value;
                }
            }
        }
        let auxC = 0;
        let con = true;
        let c = 0;
        let table = `<h2>Tabla Potencia</h2><br><table class="tabla-potencia"
border="1" cellspacing="0"> <th>\\(n\\)</th> <th> \\(x_n\\) </th> <th> \\(y_n\\)
</th> <th> \\(c_n\\) </th>`;
        let iteraciones=0;
        if(cen){
            do{
                let y = Array(n).fill(0);
                for (let i = 0; i < n; i++) {
                    for (let j = 0; j < n; j++) {
                        y[i] += matriz[i][j] * x[j];
                    }
                }
                c = 0;
                for(i=0;i<n-1;i++){
                    if(Math.abs(y[i])>c) c=y[i];
                }
                if(Math.abs(c-auxC)<0.00001){
                    con = false;
                }
            }
        }
    }

```

```

        auxC = c;
        for(i=0;i<n;i++) x[i] = c;
        for (let i = 0; i < n; i++) {
            x[i] = y[i] / c;
        }
        //Agregar los valores a la tabla
        table += `|<td width="50px">${iteraciones}</td><td
width="50px"><table class="tabla-potencia" border="1" cellspacing="0">`;
        for (let i = 0; i < n; i++) {
            table+=`<tr><td>${(x[i]).toFixed(4)}</td></tr>`;
        }
        table+=`</table></td><td width="50px"><table class="tabla-potencia"
border="1" cellspacing="0">`;
        for (let i = 0; i < n; i++) {
            table += `|<td>${(y[i]).toFixed(4)}</td></tr>`;
        }
        table += `|<td><td width="50px">${c.toFixed(4)}</td></tr>`;
        iteraciones++;
    }while(con);
    table+=`</table>`;
    let eigen = `

## Máximo valor propio</h2><p>\\(\\lambda_{max} = ${(c.toFixed(4))}\\)</p>`; let eigenV = `\\(\\vec{v}_{\\lambda} = \\begin{pmatrix}`; for(i=0;i<n;i++){ eigenV += `${(x[i]).toFixed(4)}`; if(i!=n) eigenV += '\\\\'; } eigenV += '\\end{pmatrix}\\)'; //Sobreescribir el eigenvalor y eigenvector let eigenP = document.getElementById('eigenvalor-p'); let eigenVP = document.getElementById('eigenvector-p'); let tablaP = document.getElementById('tabla-p'); eigenP.innerHTML = eigen; eigenVP.innerHTML = eigenV; tablaP.innerHTML = table; MathJax.typeset(); } } //Resolver Potencia Inversa function resolverPI() { // Leer matriz A del documento let n = parseInt(document.getElementById('orden-7').value); let cen = true; let matriz = []; let x = []; for (let i = 0; i < n; i++) { matriz[i] = []; let valorX = parseFloat(document.getElementById(`vector-${i}-p- i`).value); if (isNaN(valorX)) { alert('El valor de los coeficientes no puede estar vacío'); cen = false; break; } else { x.push(valorX); } } for (let j = 0; j < n; j++) { let value = parseFloat(document.getElementById(`matriz-${i}-${j}-p- i`).value);


|  |

|  |

|  |

```

```

        if (isNaN(value)) {
            alert('El valor de los coeficientes no puede estar vacío');
            cen = false;
            break;
        } else {
            matriz[i][j] = value;
        }
    }
}
if (!cen) return;
let matrizInversa = math.inv(matriz);
let auxC = 0;
let con = true;
let c = 0;
let table = '<h2>Tabla Potencia Inversa</h2><br><table class="tabla-potencia"
border="1" cellspacing="0"> <th>\\(n\\)</th> <th> \\(x_n\\) </th> <th> \\(y_n\\)
</th> <th> \\(c_n\\) </th>';
let iteraciones = 0;
do {
    let y = Array(n).fill(0);
    for (let i = 0; i < n; i++) {
        for (let j = 0; j < n; j++) {
            y[i] += matrizInversa[i][j] * x[j];
        }
    }
    c = Math.max(...y.map(Math.abs));
    if (Math.abs(c - auxC) < 0.00001) {
        con = false;
    }
    auxC = c;
    x = y.map(yi => yi / c);
    table += '<tr><td width="50px">${iteraciones}</td><td width="50px"><table
class="tabla-potencia" border="1" cellspacing="0">';
    for (let i = 0; i < n; i++) {
        table += '<tr><td>${x[i].toFixed(4)}</td></tr>';
    }
    table += '</table></td><td width="50px"><table class="tabla-potencia"
border="1" cellspacing="0">';
    for (let i = 0; i < n; i++) {
        table += '<tr><td>${y[i].toFixed(4)}</td></tr>';
    }
    table += '</table></td><td width="50px">${c.toFixed(4)}</td></tr>';
    iteraciones++;
} while (con);
table += '</table>';
let eigen = '<h2>Menor valor propio (en magnitud)</h2><p>\\(\\lambda_{min}
\\approx ${c.toFixed(4)}\\)</p>';
let eigenV = '\\(\\vec{v}_{\\lambda} = \\begin{pmatrix}';
for (let i = 0; i < n; i++) {
    eigenV += `${x[i].toFixed(4)} `;
    if (i !== n - 1) eigenV += '\\\\';
}
eigenV += '\\end{pmatrix}\\)';
// Actualizar el Doc
document.getElementById('eigenvalor-p-i').innerHTML = eigen;
document.getElementById('eigenvector-p-i').innerHTML = eigenV;
document.getElementById('tabla-p-i').innerHTML = table;
MathJax.typeset();
}

```

```

//SALIDA
function salida(){
    var audio = document.getElementById('sonido-salida');
    audio.play();
    document.getElementById('audio-2').play();
    document.getElementById('portada').style.display='none';
    document.getElementById('introduccion').style.display='none'
    document.getElementById('menu-principal').style.display='none'
    document.getElementById('unidad-2-introduccion').style.display='none'
    document.getElementById('unidad-2-menu').style.display='none'
    document.getElementById('falsa-posicion-1').style.display='none'
    document.getElementById('falsa-posicion-2').style.display='none'
    document.getElementById('newton-1').style.display='none'
    document.getElementById('newton-2').style.display='none'
    document.getElementById('secante-1').style.display='none'
    document.getElementById('secante-2').style.display='none'
    document.getElementById('salida').style.display='block';
    document.getElementById('unidad-3-introduccion').style.display='none';
    document.getElementById('unidad-3-menu').style.display='none';
    document.getElementById('gauss-jordan-1').style.display='none';
    document.getElementById('gauss-jordan-2').style.display='none';
    document.getElementById('jacobi-1').style.display='none';
    document.getElementById('jacobi-2').style.display='none';
    document.getElementById('gauss-seidel-1').style.display='none';
    document.getElementById('gauss-seidel-2').style.display='none';
    document.getElementById('unidad-4-introduccion').style.display='none';
    document.getElementById('unidad-4-menu').style.display='none';
    document.getElementById('doolittle-1').style.display='none';
    document.getElementById('doolittle-2').style.display='none';
    document.getElementById('cholesky-1').style.display='none';
    document.getElementById('cholesky-2').style.display='none';
    document.getElementById('unidad-5-introduccion').style.display='none';
    document.getElementById('unidad-5-menu').style.display='none';
    document.getElementById('potencia-1').style.display='none';
    document.getElementById('potencia-2').style.display='none';
    document.getElementById('potencia-inversa-1').style.display='none';
    document.getElementById('potencia-inversa-2').style.display='none';
}
window.addEventListener('load', iniciarPrograma);

```