PROYECTO FINAL

Hugo Raúl López Enriquez, Robin Israel Salazar Pinto
1690-20-2328, 1690-20-4705 Universidad Mariano Gálvez
1960-007 Algebra Lineal
hlopeze9@miumg.edu.gt , *rsalazarp1@miumg.edu.gt*

Introducción
Una matriz es un arreglo bidimensional de números, ordenados en filas y columnas, donde una fila es cada una de las líneas horizontales de la matriz y una columna es cada una de las líneas verticales. Para poder sumar o restar matrices, éstas deben de tener el mismo numero de filas y columnas, de lo contrario la operación no se puede realizar, estas operaciones son sencillas de realizar ya que solo se tiene que sumar o restar (depende del tipo de operación que se vaya a realizar) los elementos que ocupan la misma posición. En matemáticas se define el determinante como una forma multilineal alternada sobre un espacio vectorial. Esta definición indica una serie de propiedades matemáticas y generaliza el concepto de determinante de una matriz haciéndolo aplicable en numerosos campos. para poder calcular la determinante de una matriz es necesario que la matriz sea cuadrada, es decir tener el mismo número de filas y el mismo número de columnas. Estas operaciones son de las más básicas dentro del campo de matrices, pero constituyen la base para la resolución de numerosos problemas en diversos contextos.

Palabras Claves
Matrices, Suma, Resta, Determinante, Programación.

SUMA Y RESTA DE MATRICES

- **SUMA DE MATRICES**

La suma de matrices es una operación lineal que consiste en unificar los elementos de dos o más matrices que coincidan en posición dentro de sus respectivas matrices y que estas tengan el mismo orden.
En otras palabras, el sumatorio de una o más matrices es la unión de los elementos que tengan la misma posición dentro de las matrices y que estas tengan el mismo orden. (Rodó)

Fórmula para sumar matrices

$$\boldsymbol{Z}_{nxm} + \boldsymbol{X}_{nxm} + \cdots + \boldsymbol{N}_{nxm} = \begin{pmatrix} z_{11} + x_{11} + \cdots + n_{11} & \dots & z_{1m} + x_{1m} + \cdots + n_{1m} \\ \vdots & \ddots & \vdots \\ z_{n1} + x_{n1} + \cdots + n_{n1} & \dots & z_{nm} + x_{nm} + \cdots + n_{nm} \end{pmatrix}$$

Dos matrices tienen que tener un número igual de filas y columnas para poder sumarlas.[1] La suma de dos matrices **A** y **B** es una matriz que tiene el mismo número de filas y columnas que **A** y **B**. La suma de **A** y **B**, denotada como **A + B**, se computa añadiendo los elementos correspondientes de **A** y **B**:

$$A = \begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{pmatrix} \quad y \quad B = \begin{pmatrix} b_{11} & \cdots & b_{1n} \\ \vdots & \ddots & \vdots \\ b_{m1} & \cdots & b_{mn} \end{pmatrix}$$

$$A + B = C = \begin{pmatrix} c_{11} & \cdots & c_{1n} \\ \vdots & \ddots & \vdots \\ c_{m1} & \cdots & c_{mn} \end{pmatrix} \quad sería \quad c_{i,j} = a_{i,j} + b_{i,j}$$

Procedimiento

Para sumar matrices:

1. Comprobar el orden de las matrices, tal que:
- Si el orden de las matrices es el mismo, entonces se pueden sumar las matrices.
- Si el orden de las matrices es distinto, entonces no podemos sumar las matrices.
2. Sumar los elementos que tienen la misma posición dentro de sus respectivas matrices.

El sumatorio de matrices comparte las mismas características que cuando sumamos números y variables en álgebra, con la diferencia de que aquí tenemos “coordenadas”. Es decir, tendremos en cuenta la posición del elemento dentro de cada matriz. La posición de cada elemento se denota con subíndices, tal que:

$$\mathbf{a_{11}} + \mathbf{b_{11}}$$

Entonces, el sumatorio de estos dos elementos es posible dado que tienen la misma posición. En otras palabras, tienen los mismos números en los subíndices. Si la posición de los elementos fuera distinta, no podríamos sumarlos.

Ejemplo:

$$\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} + \begin{pmatrix} 5 & 6 \\ 7 & 8 \end{pmatrix} = \begin{pmatrix} 6 & 8 \\ 10 & 12 \end{pmatrix}$$

Propiedades de la Suma de Matrices

Dadas tres matrices cualquiera X, Z, Y tal que:
Matrices de orden nxm.

$$\boldsymbol{X}_{nxm} = \begin{pmatrix} x_{11} & \cdots & x_{1m} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{nm} \end{pmatrix} \quad \boldsymbol{Z}_{nxm} = \begin{pmatrix} z_{11} & \cdots & z_{1m} \\ \vdots & \ddots & \vdots \\ z_{n1} & \cdots & z_{nm} \end{pmatrix} \quad \boldsymbol{Y}_{nxm} = \begin{pmatrix} y_{11} & \cdots & y_{1m} \\ \vdots & \ddots & \vdots \\ y_{n1} & \cdots & y_{nm} \end{pmatrix}$$

- Propiedad asociativa:

Z + (X + Y) = (Z + X) + Y
Es equivalente primero sumar dos matrices y luego otra matriz al resultado anterior.

- Propiedad conmutativa:

Z + X + Y = X + Y + Z
El orden del sumatorio no es relevante.

- Elemento neutro:

Dada una matriz cero O del mismo orden que Z, X, Y, tal que:

$$\boldsymbol{O}_{nxm} = \begin{pmatrix} 0 & 0 & 0 \\ \vdots & \ddots & \vdots \\ 0 & 0 & 0 \end{pmatrix}$$

Matriz cero o nula.

Entonces, X + O = O + X = X
El efecto neutro se produce cuando sumamos la matriz objetivo con una matriz cero. El resultado es la misma matriz.

- **RESTA DE MATRICES**

La resta de matrices es una operación lineal que consiste en sustraer los elementos de dos o más matrices que coincidan en posición dentro de sus respectivas matrices y que estas tengan el mismo orden. (Rodó P.)

La resta de matrices, en definitiva, consiste en restar los distintos componentes de cada matriz, siempre respetando el lugar que ocupan en la estructura. Si las matrices tuvieran distinta cantidad de componentes, la operación no se puede completar. Sin embargo, no existe una restricción con respecto a la proporción que debe haber entre el número de filas y columnas. (Pérez Porto & Gardey, 2014)

$$A - B = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & a_{m3} & \cdots & a_{mn} \end{pmatrix} - \begin{pmatrix} b_{11} & b_{12} & b_{13} & \cdots & b_{1n} \\ b_{21} & b_{22} & b_{23} & \cdots & b_{2n} \\ b_{31} & b_{32} & b_{33} & \cdots & b_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ b_{m1} & b_{m2} & b_{m3} & \cdots & b_{mn} \end{pmatrix} =$$

$$= \begin{pmatrix} a_{11} - b_{11} & a_{12} - b_{12} & a_{13} - b_{13} & \cdots & a_{1n} - b_{1n} \\ a_{21} - b_{21} & a_{22} - b_{22} & a_{23} - b_{23} & \cdots & a_{2n} - b_{2n} \\ a_{31} - b_{31} & a_{32} - b_{32} & a_{33} - b_{33} & \cdots & a_{3n} - b_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{m1} - b_{m1} & a_{m2} - b_{m2} & a_{m3} - b_{m3} & \cdots & a_{mn} - b_{mn} \end{pmatrix}$$

Procedimiento
Para restar matrices debemos:

1. Comprobar el orden de las matrices, tal que:
- Si el orden de las matrices es el mismo, entonces **sí** se pueden restar las matrices.
- Si el orden de las matrices es distinto, entonces **no** se pueden restar las matrices.

2. Restar los elementos que tienen la misma posición dentro de sus respectivas matrices.

La diferencia de matrices comparte las mismas características que cuando restamos números y variables en álgebra, con la diferencia de que aquí tenemos "coordenadas". Es decir, tendremos en cuenta la posición del elemento dentro de cada matriz. La posición de cada elemento se denota con subíndices, tal que:

$$\mathbf{a_{11}} \; - \; \mathbf{b_{11}}$$

Posee el mismo subíndice.

Entonces, la diferencia de estos dos elementos es posible dado que tienen la misma posición. En otras palabras, tienen los mismos números en los subíndices. Si la posición de los elementos fuera distinta, no podríamos restarlos.

Ejemplo 1:

$$\begin{pmatrix} 4 & 9 \\ 2 & 7 \end{pmatrix} - \begin{pmatrix} 2 & 6 \\ -1 & 4 \end{pmatrix} = \begin{pmatrix} 2 & 3 \\ 3 & 3 \end{pmatrix}$$

Ejemplo 2:

$$A = \begin{pmatrix} 3 & 1 & 2 \\ 0 & 5 & -3 \\ 7 & 0 & 4 \end{pmatrix} \quad y \quad B = \begin{pmatrix} -1 & 2 & 4 \\ 2 & 5 & 8 \\ 0 & 1 & -2 \end{pmatrix}$$

$$A - B = \begin{pmatrix} 3-(-1) & 1-2 & 2-4 \\ 0-2 & 5-5 & -3-8 \\ 7-0 & 0-1 & 4-(-2) \end{pmatrix} = \begin{pmatrix} 4 & -1 & -2 \\ -2 & 0 & -11 \\ 7 & -1 & 6 \end{pmatrix}$$

DETERMINANTE DE UNA MATRIZ

En Matemáticas se define el determinante como una forma multilineal alternada sobre un espacio vectorial. Esta definición indica una serie de propiedades matemáticas y generaliza el concepto de determinante de una matriz haciéndolo aplicable en numerosos campos. El concepto de determinante o volumen orientado fue introducido para estudiar el número de soluciones de los sistemas de ecuaciones lineales. (Wikipedia)

Métodos de cálculo

Para el cálculo de determinantes de matrices de cualquier orden, existe una regla recursiva (teorema de Laplace) que reduce el cálculo a sumas y restas de varios determinantes de un orden inferior. Este proceso se puede repetir tantas veces como sea necesario hasta reducir el problema al cálculo de múltiples determinantes de orden tan pequeño como se quiera. Sabiendo que el determinante de un escalar es el propio escalar, es posible calcular el determinante de cualquier matriz aplicando dicho teorema.

Además de esta regla, para calcular determinantes de matrices de cualquier orden podemos usar otra definición de determinante conocida como Fórmula de Leibniz.

La fórmula de Leibniz para el determinante de una matriz cuadrada A de orden n es:

$$\begin{aligned} F(A) &= F\left(\sum_{k_1=1}^{n} a_{k_1}^{1} E^{k_1}, \ldots, \sum_{k_n=1}^{n} a_{k_n}^{n} E^{k_n}\right) \\ &= \sum_{k_1, \ldots, k_n=1}^{n} \left(\prod_{i=1}^{n} a_{k_i}^{i}\right) F\left(E^{k_1}, \ldots, E^{k_n}\right). \end{aligned}$$

La fórmula de Leibniz es útil como definición de determinante; pero, excepto para órdenes muy pequeños, no es una forma práctica de calcularlo: hay que llevar a cabo n! productos de n factores y sumar n! elementos. No se suele usar para calcular el determinante si la matriz tiene más de tres filas.

Matrices de orden inferior

El caso de matrices de orden inferior (orden 1, 2 o 3) es muy simple y su determinante se calcula con sencillas reglas conocidas. Dichas reglas son también deducibles del teorema de Laplace.

Una matriz de orden uno, es un caso trivial, pero lo trataremos para completar todos los casos. Una matriz de orden uno puede ser tratada como un escalar, pero aquí la consideraremos una matriz cuadrada de orden uno:

$$A = [a_{11}]$$

El valor del determinante es igual al único término de la matriz:

$$\det A = |\, a_{11} \,| = a_{11}$$

Una matriz de dimensión *2×2* tiene como determinante la resta del producto de los elementos de la diagonal principal con el producto de los elementos de la diagonal secundaria. El determinante de una matriz de orden 2x2 está dado por la siguiente fórmula:

$$\begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} = a_{11} \cdot a_{22} - a_{12} \cdot a_{11}$$

Dada una matriz de orden 3. El determinante de una matriz de orden 3 se calcula mediante la regla de Sarrus:

$$\begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} = \begin{array}{l} + a_{11} \cdot a_{22} \cdot a_{33} + a_{12} \cdot a_{23} \cdot a_{31} + a_{21} \cdot a_{32} \cdot a_{13} \\ - a_{31} \cdot a_{22} \cdot a_{13} - a_{32} \cdot a_{23} \cdot a_{11} - a_{21} \cdot a_{12} \cdot a_{33} \end{array}$$

Determinantes para matrices de orden superior

El determinante de orden n, puede calcularse mediante el teorema de Laplace a partir de una fila o columna, reduciendo el problema al cálculo de n determinantes de orden n-1. Para ello se toma una fila o columna cualquiera, multiplicando cada elemento por su cofactor. El cofactor de un elemento a_{ij} de la matriz es el determinante de la matriz que se obtiene al eliminar la fila y columna correspondiente a dicho elemento, y multiplicándolo por $(-1)^{i+j}$, donde i es el número de fila y j el número de columna.

La suma de todos los productos de los elementos de una fila (o columna) cualquiera multiplicados por sus cofactores es igual al determinante.

En caso de un determinante de orden 4, se obtienen directamente determinantes de orden 3 que podrán ser calculados por la regla de Sarrus. En cambio, en los determinantes de orden superior, como por ejemplo n = 5, al desarrollar los elementos de una línea, obtendremos determinantes de orden 4, que a su vez se deberán desarrollar en por el mismo método, para obtener determinantes de orden 3. Por ejemplo, para obtener con el método especificado un determinante de orden 4, se deben calcular 4 determinantes de orden 3. Esto puede aligerarse si previamente se logran tres ceros en una fila o columna, bastando entonces con calcular un determinante de orden 3 (ya que los demás determinantes estarán multiplicados por 0, lo que los anula).

La cantidad de operaciones aumenta muy rápidamente. Por ejemplo, mediante este método, para un determinante de orden 10 se deberán calcular 10 x 9 x 8 x 7 x 6 x 5 x 4 = 604.800 determinantes de orden 3.

También puede utilizarse el Método de eliminación Gaussiana, para convertir la matriz en una matriz triangular. (Wikipedia, Wikipedia)

SOFTWARE MATLAB

MATLAB (abreviatura de MATrix LABoratory, “laboratorio de matrices”) es un sistema de cómputo numérico que ofrece un entorno de desarrollo integrado (IDE) con un lenguaje de programación propio (lenguaje M). Está disponible para las plataformas Unix, Windows, macOS y GNU/Linux.

Entre sus prestaciones básicas se hallan la manipulación de matrices, la representación de datos y funciones, la implementación de algoritmos, la creación de interfaces de usuario (GUI) y la comunicación con programas en otros lenguajes y con otros dispositivos hardware. El paquete MATLAB dispone de dos herramientas adicionales que expanden sus prestaciones, a saber, Simulink (plataforma de simulación multidominio) y GUIDE (editor de interfaces de usuario - GUI). Además, se pueden ampliar las capacidades de MATLAB con las cajas de herramientas (toolboxes); y las de Simulink con los paquetes de bloques (blocksets).

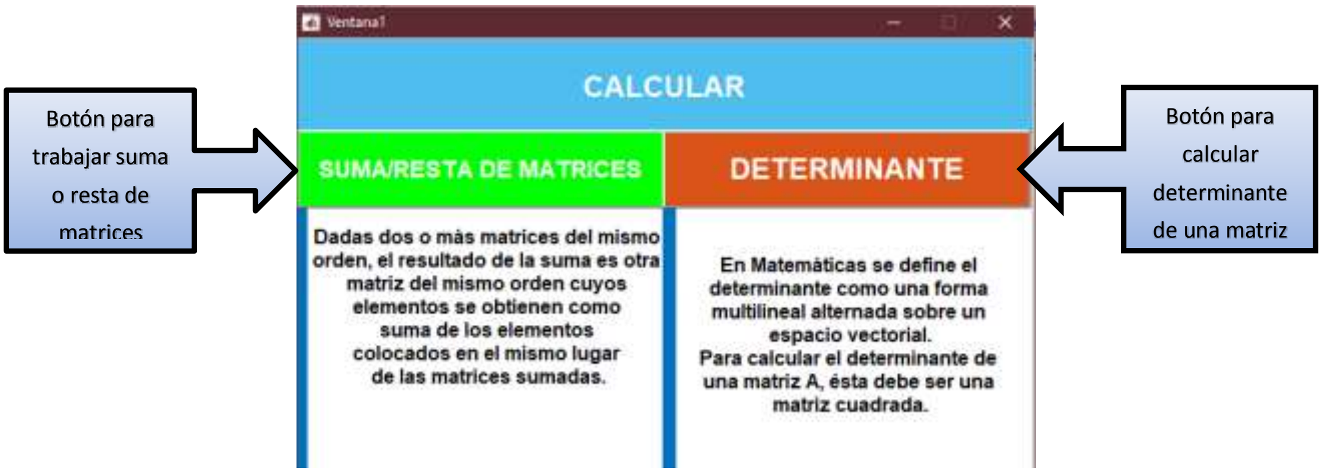
Es un software muy usado en universidades y centros de investigación y desarrollo. En los últimos años ha aumentado el número de prestaciones, como la de programar directamente procesadores digitales de señal o crear código VHDL.

En 2004, se estimaba que MATLAB era empleado por más de un millón de personas en ámbitos académicos y empresariales. Las aplicaciones de MATLAB se desarrollan en un lenguaje de programación propio. Este lenguaje es interpretado, y puede ejecutarse tanto en el entorno interactivo, como a través de un archivo de script (archivos *.m). Este lenguaje permite operaciones de vectores y matrices, funciones, cálculo lambda, y programación orientada a objetos. (Wikipedia, MATLAB)

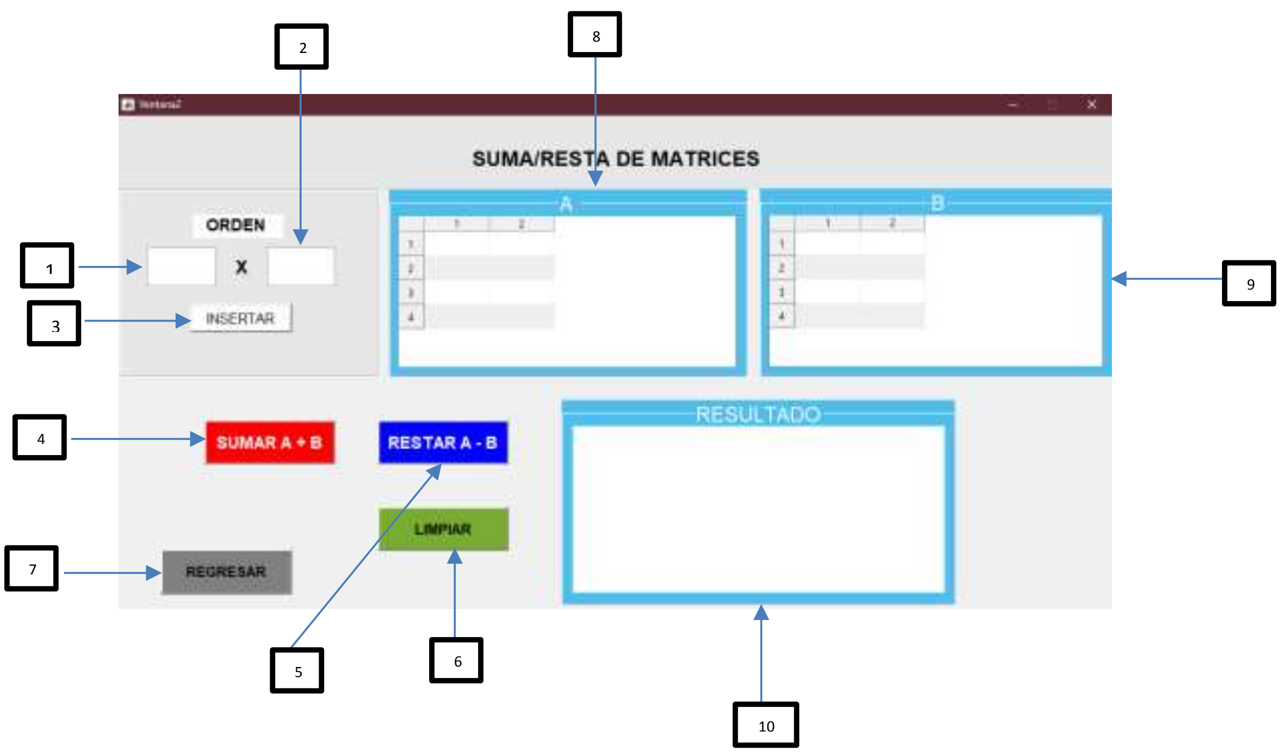
MANUAL TECNICO

- **INTERFAZ**

VENTANA PRINCIPAL (Ventana1)

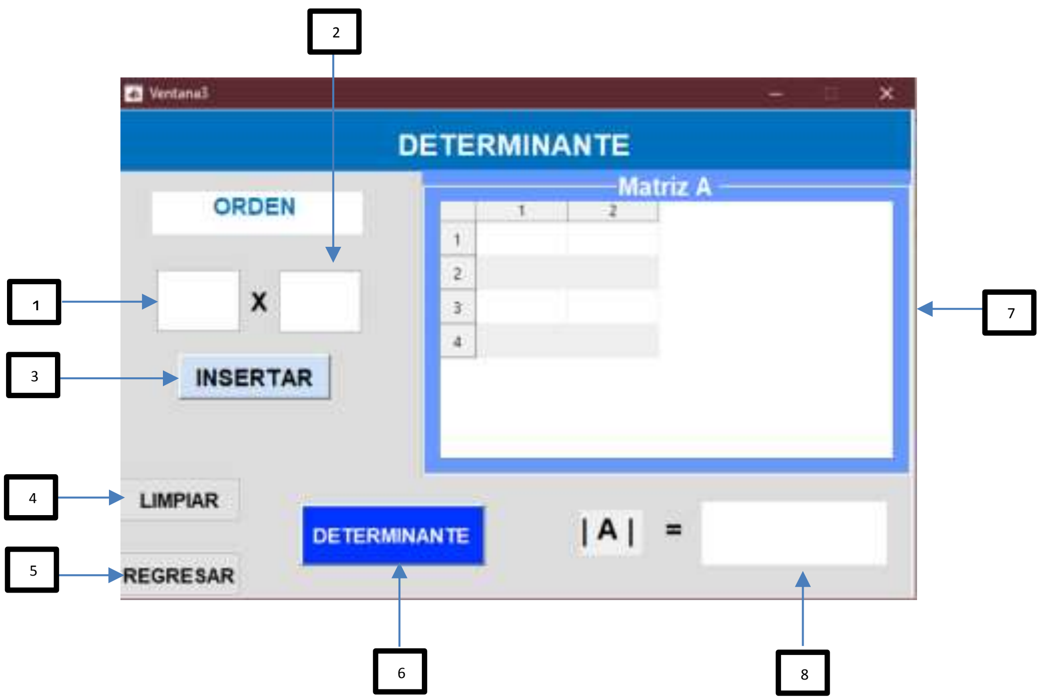


VENTANA DE "SUMA Y RESTA DE MATRICES" (Ventana2)



1. Casilla para ingresar el número de filas de las matrices que van a trabajar.
2. Casilla para ingresar el número de columnas de las matrices que van a trabajar.
3. Botón para crear las tablas con el tamaño que se ingresó.
4. Botón para que las matrices se sumen.
5. Botón para que las matrices se resten.
6. Botón para limpiar los datos ingresados.
7. Botón para regresar a la ventana de inicio.
8. Tabla de matriz “A”.
9. Tabla de matriz “B”.
10. Cuadro de texto donde se mostrará el resultado.

VENTANA “DETERMINANTE” (Ventana3)



1. Casilla para ingresar el número de filas de la matriz que se quiere trabajar.
2. Casilla para ingresar el número de columnas de la matriz que se quiere trabajar.
3. Botón para crear la tabla con el orden ingresado.
4. Botón para borrar los datos ingresados.
5. Botón para regresar a la ventana principal.
6. Botón para hacer el cálculo de la determinante.
7. Tabla para ingresar los datos de la matriz.
8. Casilla donde se mostrará el determinante de la matriz ingresada.

- **CODIGO**

VENTANA PRINCIPAL (Ventana1)

La ventana principal solamente posee dos botones, acontinuación se dan a conocer las funciones de dichos botones.

- **Función Botón "SUMA/RESTA DE MATRICES" (sumaResta)**

```
% --- Executes on button press in sumaResta.
%Funcion del boton "Suma/Resta de Matrices
%Se cierra la ventana actual "Ventana1" y se abre la Ventana2
function sumaResta_Callback(hObject, eventdata, handles)
close(Ventana1)
Ventana2
```

La función del botón "sumaResta" posee únicamente dos líneas de código, en la primera con "close()" se cierra la ventana que se ingrese como parámetro, como se muestra en el código se cierra la ventana actual "Ventana1".
La segunda línea de código, nos indica la ventana que se abrirá luego de cerrar la actual, es decir la Ventana2 para calcular sumas y restas de matrices.

- **Función Botón "DETERMINANTE" (determ)**

```
% --- Executes on button press in determ.
%Funcion del boton "DETERMINANTE"
%Cierra la ventana actual "Ventana1" y abre la Ventana3
function determ_Callback(hObject, eventdata, handles)
close(Ventana1)
Ventana3
```

La función del botón "determ" tiene únicamente dos instrucciones, en la primera se cierra la ventana actual utilizando ""close()" y pasamos como parámetro el nombre de la ventana.
En la siguiente instrucción se indica el nombre de la ventana que se abrirá, es decir "Ventana3" para el cálculo de Determinante de una matriz.

VENTANA DE “SUMA Y RESTA DE MATRICES” (Ventana2)

La ventana para calcular sumas y restas de matrices posee cinco botones, a continuación se expone las funciones de los botones y las instrucciones de éstas.

- **Función Botón “INSERTAR” (insertar)**

```
% --- Executes on button press in insertar.
% Función del Boton Insertar
function insertar_Callback(hObject, eventdata, handles)
% Se obtiene el numero de filas y el numero de columnas.
fila = str2double(get(handles.filas, 'String'));
columna = str2double(get(handles.columnas, 'String'));
% Se condiciona que si las variables fila o columna obtienen valores que no
% son numericos se muestre una advertencia.
if isnan(fila) || isnan(columna)
    warndlg('Se deben ingresar números para el orden de las matrices.', 'Advertencia');
% De lo contrario se procede a establecer el orden ingresado a la matrizA y
% a la matrizB.
else
    orden = cell (fila, columna);
    orden(:,:)={''};
    set(handles.matrizA, 'Data', orden);
    set(handles.matrizA, 'ColumnEditable', true(1, columna));
    set(handles.matrizB, 'Data', orden);
    set(handles.matrizB, 'ColumnEditable', true(1, columna));
end
```

La función del botón “insertar” posee una serie de instrucciones para establecer el orden ingresado por el usuario a la Matriz A y a la Matriz B.

En las primeras instrucciones se utiliza “get” para conseguir los números ingresados por el usuario, con “str2double” pasamos el string obtenido a datos numéricos, si dicho string está vacío o es un carácter que no corresponde a un número el valor resultante será “NaN”, por último se almacenan en las variables “fila” y “columna” respectivamente.

Luego, usando la estructura “if” se condiciona el enviar una advertencia si los valores de las variables que almacenan el número de filas y columnas son NaN. En caso contrario, se procede a crear una matriz de celdas llamada “orden” usando la función “cell()” y pasando como parámetros el número de filas y de columnas, se establecen todas las filas y columnas sin ningún carácter.

Por último utilizando “set” se establece la matriz con el orden ingresado en las dos tablas, matrizA y matrizB, además se hace editable desde la columna 1 hasta la última columna de las matrices.

- **Función Botón "SUMAR A + B" (sumar)**

```
% --- Executes on button press in sumar.
% Funcion Boton Sumar A+B
function sumar_Callback(hObject, eventdata, handles)
% Se obtienen el numero de filas y columnas ingresados.
fila = str2double(get(handles.filas, 'String'));
columna = str2double(get(handles.columnas, 'String'));
% Se condiciona que si las variables fila o columna obtienen valores que no
% son numericos se muestre una advertencia.
if isnan(fila) || isnan(columna)
    warndlg('Se deben ingresar números para el orden de las matrices.', 'Advertencia');
%De caso contrario se obtiene los valores ingresados en la matriz A y B se
%crea la matriz R que contiene el resultado de sumar A+B y se muestra.
else
    A = get(handles.matrizA, 'data');
    A = str2double(A);
    B = get(handles.matrizB, 'data');
    B = str2double(B);
    R = A + B;
    set(handles.resultado, 'String', num2str(R));
end
```

La función del botón llamado "sumar" posee las siguientes instrucciones:

Utiliza "get" para conseguir los números que conforman el orden de la matriz, los convierte en datos númericos usando "str2double", si dicho string está vacío o es un carácter que no corresponde a un número el valor resultante será "NaN", después los guarda en las variables "fila" y "columna".

Luego usando "if" y con la función "isnan()" se condiciona el enviar una advertencia si los valores de las variables que almacenan el número de filas y columnas son NaN. En caso contrario usando "get" conseguimos los datos ingresados en las dos tablas que corresponden a las dos matrices, y se guardan en dos variables, A y B, usando "str2double" se convierten los datos de "string" a "double".

Por último se suman la matriz A y la matriz B y se guarda en la matriz R, usando "set" establecemos el resultado, es decir la matriz R en el cuadro de texto "resultado".

- **Función Botón "RESTAR A - B" (restar)**

```
% --- Executes on button press in restar.
% Funcion Boton Restar A-B
function restar_Callback(hObject, eventdata, handles)
% Se obtienen el numero de filas y columnas ingresados.
fila = str2double(get(handles.filas, 'String'));
columna = str2double(get(handles.columnas, 'String'));
% Se condiciona que si las variables fila o columna obtienen valores que no
% son numericos se muestre una advertencia.
if isnan(fila) || isnan(columna)
    warndlg('Se deben ingresar números para el orden de las matrices.', 'Advertencia');
%De caso contrario se obtiene los valores ingresados en la matriz A y B se
%crea la matriz R que contiene el resultado de restar A-B y se muestra.
else
    A = get(handles.matrizA, 'data');
    A = str2double(A);
    B = get(handles.matrizB, 'data');
    B = str2double(B);
    R = A - B;
    set(handles.resultado, 'String', num2str(R));
end
```

La función del botón llamado "restar" posee las siguientes instrucciones:

Utiliza "get" para conseguir los números que conforman el orden de la matriz, los convierte en datos númericos usando "str2double" , si dicho string está vacío o es un carácter que no corresponde a un número el valor resultante será "NaN", despúes los guarda en las variables "fila" y "columna".

Luego usando "if" y con la función "isnan()" se condiciona el enviar una advertencia si los valores de las variables que almacenan el número de filas y columnas son NaN. En caso contrario usando "get" conseguimos los datos ingresados en las dos tablas que corresponden a las dos matrices, y se guardan en dos variables, A y B, usando "str2double" se convierten los datos de "string" a "double".

Por último se restan la matriz A y la matriz B y se guarda en la matriz R, usando "set" establecemos el resultado, es decir la matriz R en el cuadro de texto "resultado".

- **Función Botón “LIMPIAR”**

```
% --- Executes on button press in limpiar.
% Funcion Boton Limpiar
function limpiar_Callback(hObject, eventdata, handles)
%LIMPIAR LOS CUADROS DE TEXTO
set(handles.filas, 'String', '');
set(handles.columnas, 'String', '');
set(handles.resultado, 'String', '');
```

La función del botón “limpiar” únicamente posee tres instrucciones, y éstas son las de establecer por medio del método “set” ningún carácter (‘ ’) en los cuadros de texto “filas”, “columnas” y “resultado”.

- **Función Botón “REGRESAR”**

```
% --- Executes on button press in pushbutton6.
% Funcion boton Regresar
function pushbutton6_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton6 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
close(Ventana2)
Ventana1
```

La función del botón posee dos pequeñas instrucciones, la primera utiliza la función “close()” para cerrar la ventana actual, Ventana2, y en la segunda instrucción se indica la ventana que se abrirá luego de cerrar la actual. Como el boton menciona, se pretende regresar, por lo que se coloca el nombre de la ventana inicial, Ventana1.

VENTANA “DETERMINANTE” (Ventana3)

La ventana para calcular el determinante de una matriz posee cuatro botones, a continuación se expone las funciones de los botones y las instrucciones de éstas.

- **Función Botón “INSERTAR” (insertar)**

```
% --- Executes on button press in insertar.
% Funcion Boton Insertar
function insertar_Callback(hObject, eventdata, handles)
% Se obtiene el numero de filas y el numero de columnas.
fila = str2double(get(handles.filas, 'String'));
columna = str2double(get(handles.columnas, 'String'));
% Se condiciona que si las variables fila o columna obtienen valores que no
% son numericos se muestre una advertencia.
if isnan(fila) || isnan(columna)
    warndlg('Se deben ingresar números para el orden de las matrices.', 'Advertencia');
else
    %En caso contrario, se condiciona que si el numero de filas no es igual
    %al numero de columnas se muestre un mensaje de advertencia.
    if (fila ~= columna)
        warndlg('Para calcular |A|, la matriz debe ser cuadrada.', 'Advertencia');
    %En caso contrario, se procede a establecer el orden de la matrizA.
    else
        orden = cell (fila, columna);
        orden(:,:)={''};
        set(handles.matrizA, 'Data', orden);
        set(handles.matrizA, 'ColumnEditable', true(1, columna));
    end
end
```

La función del botón “insertar” posee una serie de instrucciones para establecer el orden ingresado por el usuario a la Matriz A.

En las primeras instrucciones se utiliza “get” para conseguir los números ingresados por el usuario, con “str2double” pasamos el string obtenido a datos numéricos, si dicho string está vacío o es un carácter que no corresponde a un número el valor resultante será “NaN”, por último se almacenan en las variables “fila” y “columna” respectivamente.

Luego, usando la estructura “if” se condiciona el enviar una advertencia si los valores de las variables que almacenan el número de filas y columnas son NaN. En caso contrario, se condiciona que si el número de filas no es igual al número de columnas ingresado, se notifique al usuario que para calcular un determinante la matriz debe ser cuadrada; en caso contrario utilizando la función “cell()” se crea una matriz de celdas y se establece con el método “set” dicha matriz en la tabla llamada “matrizA”, además se establecen editables todas las celdas de columna 1 a la final.

- **Función Botón "DETERMINANTE" (determinante)**

```
function determinante_Callback(hObject, eventdata, handles)
% Se obtiene el numero de filas y el numero de columnas.
fila = str2double(get(handles.filas, 'String'));
columna = str2double(get(handles.columnas, 'String'));
% Se condiciona que si las variables fila o columna obtienen valores que no
% son numericos se muestre una advertencia.
if isnan(fila) || isnan(columna)
    warndlg('Se deben ingresar números para el orden de las matrices.', 'Advertencia');
else
    %En caso contrario, se condiciona que si el numero de filas no es igual
    %al numero de columnas se muestre un mensaje de advertencia.
    if (fila ~= columna)
        warndlg('Para calcular |A|, la matriz debe ser cuadrada.', 'Advertencia');
    % En caso contrario se obtiene la matriz ingresada, se convierte a
    % datos numericos, se calcula su determinante, se guarda en la variable
    % R y se muestra en el cuadro de texto "respuesta".
    else
        A = get(handles.matrizA, 'data');
        A = str2double(A);
        R = det(A);
        set(handles.respuesta, 'String', num2str(R));
    end
end
```

La función del botón "determinante" posee una serie de instrucciones para establecer el orden ingresado por el usuario a la Matriz A.

En las primeras instrucciones se utiliza "get" para conseguir los números ingresados por el usuario, con "str2double" pasamos el string obtenido a datos numéricos, si dicho string está vacío o es un carácter que no corresponde a un número el valor resultante será "NaN", por último se almacenan en las variables "fila" y "columna" respectivamente.

Luego, usando la estructura "if" se condiciona el enviar una advertencia si los valores de las variables que almacenan el número de filas y columnas son NaN. En caso contrario, se condiciona con "if" que si el número de filas no es igual al número de columnas ingresado, se notifique al usuario que para calcular un determinante la matriz debe ser cuadrada; en caso contrario se crea la matriz A con los valores obtenidos de la tabla "matrizA" por medio del método "get", usando el método "str2double" se convierte la matriz en datos numéricos y por último en la variable R se guarda el determinante de la matriz A obtenido haciendo uso del método "det()" y se muestra en el cuadro de texto "respuesta".

- **Función Botón “LIMPIAR”**

```
% --- Executes on button press in limpiar.
% Funcion Boton Limpiar
function limpiar_Callback(hObject, eventdata, handles)
% Limpia los cuadros de texto.
set(handles.filas, 'String', '');
set(handles.columnas, 'String', '');
set(handles.respuesta, 'String', '');
```

La función del botón “limpiar” únicamente posee tres instrucciones, y éstas son las de establecer por medio del método “set” ningún carácter (‘ ’) en los cuadros de texto “filas”, “columnas” y “respuesta”.

- **Función Botón “REGRESAR”**

```
% --- Executes on button press in regresar.
% Funcion Boton Regresar
function regresar_Callback(hObject, eventdata, handles)
close(Ventana3)
Ventana1
```

La función del botón “regresar” posee dos pequeñas instrucciones, la primera utiliza la función “close()” para cerrar la ventana actual, Ventana3, y en la segunda instrucción se indica la ventana que se abrirá luego de cerrar la actual. Como el boton menciona, se pretende regresar, por lo que se coloca el nombre de la ventana inicial, Ventana1.

Observaciones y comentarios

Son temas muy interesantes y fáciles de entender, estas nos pueden ser de utilidad en el transcurso de nuestra carrera, ya que en programación se hace uso de ellas para el manejo de información, realmente las matrices tienen muchos usos, se utilizan en la electrónica, electricidad, informática, música, y muchas otras áreas. Matlab es un software muy útil, con funciones muy prácticas y de gran importancia que ayudan a la elaboración de programas orientados a la solución de problemas de índole matemática.

Conclusiones

Matrices es un tema muy amplio y de vital importancia, éstas tienen diversas aplicaciones en la vida en general, es por ello que debemos de conocer a fondo los temas expuestos en la presente documentación. El orden de una matriz es la dimensión que ésta posee, es decir su número de filas y de columnas, para sumar y restar matrices es necesario que éstas tengan el mismo orden para que cada elemento de una matriz pueda sumarse o restarse con otro que posee el mismo subíndice de la otra matriz. Ahora bien, el determinante de una matriz es utilizado en distintos teoremas matemáticos y puede ser calculado por medio distintos métodos. Matlab es un software utilizado para la creación de programas que utilizan cálculos matemáticos, brinda opciones para creación de interfaz gráfica y distintas funciones matemáticas ya integradas.

Referencias Bibliográficas

Pérez Porto , J., & Gardey, A. (2014). *DefinicionDe*. Recuperado el 10 de Octubre de 2020, de Resta de Matrices: https://definicion.de/resta-de-matrices/

Rodó, P. *Economipedia*. Recuperado el 10 de Octubre de 2020, de Suma de Matrices: https://economipedia.com/definiciones/suma-de-matrices.html

Rodó, P. *Economipedia*. Recuperado el 10 de Octubre de 2020, de Resta de Matrices: https://economipedia.com/definiciones/resta-de-matrices.html

Wikipedia. Recuperado el 12 de Octubre de 2020, de MATLAB: https://es.wikipedia.org/wiki/MATLAB

Wikipedia. Recuperado el 10 de Octubre de 2020, de Determinante: https://es.wikipedia.org/wiki/Determinante_(matem%C3%A1tica)

Wikipedia. Recuperado el 10 de Octubre de 2020, de Determinante: https://es.wikipedia.org/wiki/Determinante_(matem%C3%A1tica)#M%C3%A9todos_de_c%C3%A1lculo