

# SUMA, RESTA, DETERMINANTES DE MATRICES Y PRODUCTO ESCALAR

1690 -20- 5944 L. Y. Y. Juárez Aldana

1690 -20- 6107 C. C. A. López Aguilar

Universidad Mariano Gálvez

1690-007 Algebra Lineal

[Ljuarez8@miumg.edu.gt](mailto:Ljuarez8@miumg.edu.gt)

[Clopeza35@miumg.edu.gt](mailto:Clopeza35@miumg.edu.gt)

## Introducción

La calculadora cuenta con cinco paneles, seis botones, cinco textos, treinta y tres editores de texto, donde se desarrollan las operaciones que la calculadora puede realizar. Se hizo uso de los comandos que tiene Matlabr2015 para las operaciones de matrices y vectores. La calculadora opera únicamente matrices de orden 3x3. La suma y resta de la matriz A y la matriz B se muestran en la matriz C. El resultado de las determinantes y el producto escalar se muestran en cuadros de texto. Además, uno de los botones limpia los datos ingresados dentro de la calculadora.

## Software

```
%Definimos la posición de cada editor de texto dentro de las tres matrices.
```

```
function C11_Callback(hObject, ~, ~)
global C;
C(1,1)=str2double(get(hObject, 'String'));
```

```
unction C21_Callback(hObject, ~, ~)
global C;
C(2,1)=str2double(get(hObject, 'String'));
```

```
function C31_Callback(hObject, ~, ~)
global C;
C(3,1)=str2double(get(hObject, 'String'));
```

```
function C22_Callback(hObject, ~, ~)
global C;
C(2,2)=str2double(get(hObject, 'String'));
```

```
function C32_Callback(hObject, ~, ~)
global C;
C(3,2)=str2double(get(hObject, 'String'));
```

```
function C13_Callback(hObject, ~, ~)
global C;
C(1,3)=str2double(get(hObject, 'String'));
```

```
function C23_Callback(hObject, ~, ~)
global C;
C(2,3)=str2double(get(hObject, 'String'));
```

```
function C33_Callback(hObject, ~, ~)
global C;
C(3,3)=str2double(get(hObject, 'String'));
```

```
function C12_Callback(hObject, ~, ~)
global C;
C(1,2)=str2double(get(hObject, 'String'));
```

```
function B11_Callback(hObject, ~, ~)
global B;
B(1,1)=str2double(get(hObject, 'String'));
```

```
function B21_Callback(hObject, ~, ~)
global B;
B(2,1)=str2double(get(hObject, 'String'));
```

```
function B31_Callback(hObject, ~, ~)
global B;
B(3,1)=str2double(get(hObject, 'String'));
```

```
function B22_Callback(hObject, ~, ~)
global B;
B(2,2)=str2double(get(hObject, 'String'));
```

```
function B32_Callback(hObject, ~, ~)
global B;
B(3,2)=str2double(get(hObject, 'String'));
```

```
function B13_Callback(hObject, ~, ~)
global B;
B(1,3)=str2double(get(hObject, 'String'));
```

```
function B23_Callback(hObject, ~, ~)
global B;
B(2,3)=str2double(get(hObject, 'String'));
```

```
function B33_Callback(hObject, ~, ~)
global B;
B(3,3)=str2double(get(hObject, 'String'));
```

```
function B12_Callback(hObject, ~, ~)
global B;
B(1,2)=str2double(get(hObject, 'String'));
```

```
function A11_Callback(hObject, ~, ~)
global A;
A(1,1)=str2double(get(hObject, 'String'));
```

```
function A21_Callback(hObject, ~, ~)

global A;
A(2,1)=str2double(get(hObject, 'String'));
```

```
function A31_Callback(hObject, ~, ~)
global A;
A(3,1)=str2double(get(hObject, 'String'));
```

```
function A22_Callback(hObject, ~, ~)
global A;
A(2,2)=str2double(get(hObject, 'String'));
```

```
function A32_Callback(hObject, ~, ~)
global A;
A(3,2)=str2double(get(hObject, 'String'));
```

```
function A13_Callback(hObject, ~, ~)
global A;
A(1,3)=str2double(get(hObject, 'String'));
```

```
function A23_Callback(hObject, ~, ~)
global A;
A(2,3)=str2double(get(hObject, 'String'));
```

```
function A33_Callback(hObject, ~, ~)
global A;
A(3,3)=str2double(get(hObject, 'String'));
```

```
function A12_Callback(hObject, ~, ~)
global A;
A(1,2)=str2double(get(hObject, 'String'));
```

```

% Se usó el comando det para buscar las determinantes, de esta manera se
% evitó realizar manualmente cada método para buscar cada determinante
function detA_Callback(~, ~, handles)
global A;
A= det (A);
set(handles.deter_A, 'string', A)

```

```

% Se usó el comando det para buscar las determinantes, de esta manera se
% evitó realizar manualmente cada método para buscar cada determinante
function detB_Callback(~, ~, handles)
global B;
B= det (B);
set(handles.deter_B, 'string', B)

```

```

function botLimpiar_Callback(~, ~, handles)

```

```

%Se borran los datos ingresados en la matriz A

```

```

set(handles.A11, 'string', ' ');
set(handles.A12, 'string', ' ');
set(handles.A13, 'string', ' ');
set(handles.A21, 'string', ' ');
set(handles.A22, 'string', ' ');
set(handles.A23, 'string', ' ');
set(handles.A31, 'string', ' ');
set(handles.A32, 'string', ' ');
set(handles.A33, 'string', ' ');

```

```

%Se borran los datos ingresados en la matriz B

```

```

set(handles.B11, 'string', ' ');
set(handles.B12, 'string', ' ');
set(handles.B13, 'string', ' ');
set(handles.B21, 'string', ' ');
set(handles.B22, 'string', ' ');
set(handles.B23, 'string', ' ');
set(handles.B31, 'string', ' ');
set(handles.B32, 'string', ' ');
set(handles.B33, 'string', ' ');

```

```

%Se borran los datos ingresados en la matriz C

```

```

set(handles.C11, 'string', ' ');
set(handles.C12, 'string', ' ');
set(handles.C13, 'string', ' ');
set(handles.C21, 'string', ' ');
set(handles.C22, 'string', ' ');
set(handles.C23, 'string', ' ');
set(handles.C31, 'string', ' ');
set(handles.C32, 'string', ' ');
set(handles.C33, 'string', ' ');

```

```

%Se borran los datos ingresados en el vector u

```

```

set(handles.U11, 'string', ' ');
set(handles.U12, 'string', ' ');
set(handles.U13, 'string', ' ');

```

```

%Se borran los datos ingresados en el vector u
set(handles.V11, 'string', ' ');
set(handles.V12, 'string', ' ');
set(handles.V13, 'string', ' ');

%Se borran los datos que muestran la determinante A
set(handles.deter_A, 'string', ' ');

%Se borran los datos que muestran la determinante B
set(handles.deter_B, 'string', ' ');

%Se borran los datos ingresados en el vector u
set(handles.escalar, 'string', ' ');

%Este Botón activa el comando suma el cual ejecuta que todos los numeros
%escritos en la matriz A y B se sumen y den el resultado en la matriz C
function botSuma_Callback(~, ~, handles)
global A;
global B;
global C;

C=A+B;

set(handles.C11, 'string',C(1,1));
set(handles.C12, 'string',C(1,2));
set(handles.C13, 'string',C(1,3));
set(handles.C21, 'string',C(2,1));
set(handles.C22, 'string',C(2,2));
set(handles.C23, 'string',C(2,3));
set(handles.C31, 'string',C(3,1));
set(handles.C32, 'string',C(3,2));
set(handles.C33, 'string',C(3,3));

%Este Botón activa el comando resta el cual ejecuta que todos los numeros
%escritos en la matriz A y B se resten y den el resultado en la matriz C
function botResta_Callback(~, ~, handles)

global A;
global B;
global C;

C=A-B;

set(handles.C11, 'string',C(1,1));
set(handles.C12, 'string',C(1,2));
set(handles.C13, 'string',C(1,3));
set(handles.C21, 'string',C(2,1));
set(handles.C22, 'string',C(2,2));
set(handles.C23, 'string',C(2,3));
set(handles.C31, 'string',C(3,1));
set(handles.C32, 'string',C(3,2));
set(handles.C33, 'string',C(3,3));

```

%Este Botón activa el comando escalar el cual ejecuta que todos los números  
%escritos en el vector U y V se multipliquen y sumen y den el resultado de un  
entero

```
function botEscalar_Callback(~, ~, handles)
```

```
global u;  
global v;  
global escalar;
```

```
escalar = dot(u,v);
```

```
set(handles.escalar, 'string', escalar);
```

%Definimos la posición de los vectores

```
function V12_Callback(hObject, ~, ~)  
global v;  
v(1,2)=str2double(get(hObject, 'String'));
```

```
function V11_Callback(hObject, ~, ~)  
global v;  
v(1,1)=str2double(get(hObject, 'String'));
```

```
function V12_Callback(hObject, ~, ~)  
global v;  
v(1,2)=str2double(get(hObject, 'String'));
```

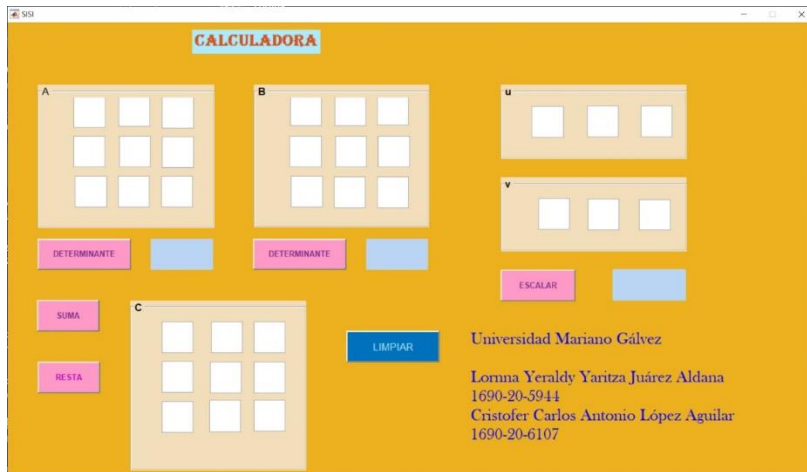
```
function V13_Callback(hObject, ~, ~)  
global v;  
v(1,3)=str2double(get(hObject, 'String'));
```

```
function U11_Callback(hObject, ~, ~)  
global u;  
u(1,1)=str2double(get(hObject, 'String'));
```

```
function U12_Callback(hObject, ~, ~)  
global u;  
u(1,2)=str2double(get(hObject, 'String'));
```

```
function U13_Callback(hObject, ~, ~)  
global u;  
u(1,3)=str2double(get(hObject, 'String'));
```

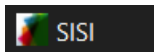
# Manual



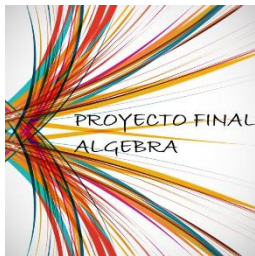
## Pasos Para el uso de la Calculadora

### ➤ Paso 1.

Ejecutar el Programa

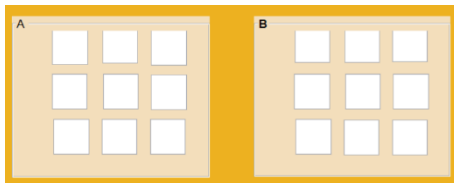


Les desplegará una ventana de Bienvenida

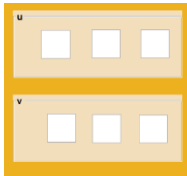


### ➤ Paso 2.

Ya una vez dentro de la Calculadora, le desplegara dos paneles. Uno para ingresar datos en el panel de matrices

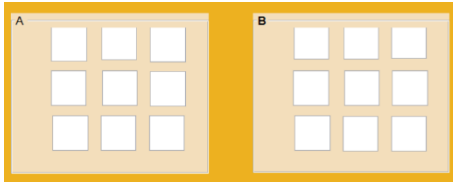


O si desea ingresar en el otro panel de vectores,



➤ Paso 3.

Si seleccionó el panel de matrices, Ingresar los datos a utilizar



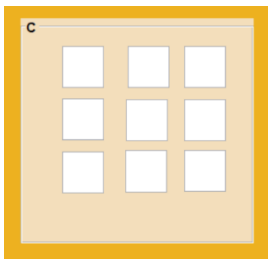
➤ Paso 4.

Seleccione la Función que va a utilizar



➤ Paso 5.

Una vez hecha la función deseada, el resultado de la operación la desplegará en el panel C



➤ Paso 6.

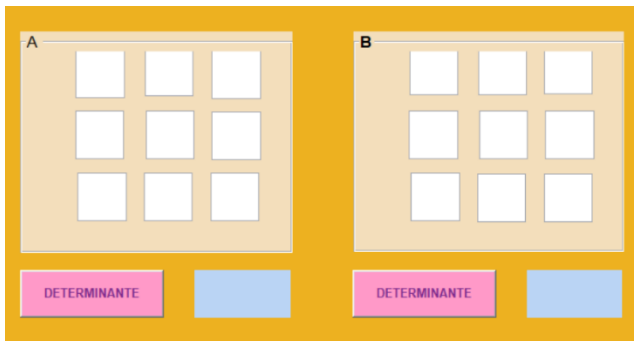
Si desea eliminar la información, oprima el botón LIMPIAR para que la información escrita se elimine



➤ Paso 7.

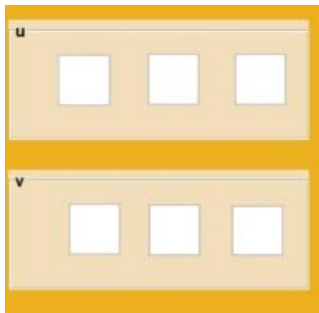


Luego si desea realizar la determinante de cada Matriz, Presionar los Botones de determinantes que se muestran debajo de cada panel de matrices.



➤ Paso 8.

Si seleccionó el panel de Vectores, Ingresar los datos a utilizar



➤ Paso 9.

Oprima el botón Escalar para obtener el resultado de la operación.



➤ Paso 10.

Si desea eliminar la información, oprima el botón LIMPIAR para que la información escrita se elimine

