

**UNIVERSIDAD TECNOLÓGICA NACIONAL
FACULTA REGIONAL ROSARIO**

FUNDAMENTOS DE INFORMÁTICA

Profesor: Dra. Sonia Benz.

Auxiliares: Ing. Evangelina Delfratte - Ing. Patricia Mores

**Introducción a la programación.
Implementación de algoritmos simples en Matlab.**

Elaboración: Ing. Patricia Mores - Ing. Evangelina Delfratte

TABLA DE CONTENIDOS

PROGRAMACIÓN.	3
IMPLEMENTACIÓN DE ALGORITMOS CUANTITATIVOS EN MATLAB.	3
COMANDOS BÁSICOS.	4
ETAPAS PARA LA SOLUCIÓN DE UN PROBLEMA UTILIZANDO MATLAB.	5
EJEMPLOS DE APLICACIÓN.	7
EJEMPLOS CON ESTRUCTURA SECUENCIAL PURA.	7
EJEMPLOS CON ESTRUCTURAS DE SELECCIÓN/ DECISIÓN.	8
EJEMPLOS CON ESTRUCTURAS DE REPETICIÓN.	15
EJEMPLOS CON ESTRUCTURAS ANIDADAS.	23

PROGRAMACIÓN.

Como se mencionó hasta ahora, un algoritmo representa solo el planteo del problema y una posible solución. Cuando un algoritmo es implementado en cualquier lenguaje de programación, reflejando las ideas desarrolladas en la etapa de análisis y diseño del algoritmo, se está creando un programa y será necesaria una computadora (equipo) para su ejecución.

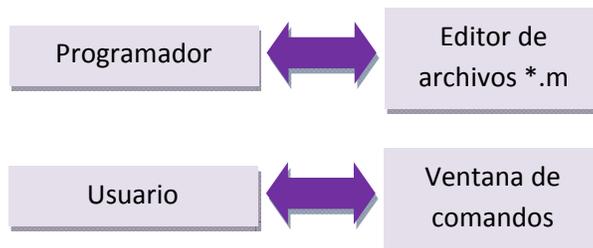


A continuación se listan todas las etapas que llevan a la solución de un determinado problema mediante programación.

- Análisis del problema, definición y delimitación.
- Diseño y desarrollo del algoritmo (diagramas de flujo, pseudocódigo, etc.).
- Prueba de escritorio. El algoritmo debe seguirse paso a paso verificando que se realicen todas las instrucciones necesarias para alcanzar el objetivo.
- Codificación. Selección del lenguaje de programación. Escritura del algoritmo utilizando la sintaxis y estructura gramatical del lenguaje seleccionado.
- Compilación. Transformación del lenguaje de programación en lenguaje de máquina.
- Depuración (debug). Proceso de detección y eliminación de los errores de programación.
- Evaluación de resultados. Se debe ejecutar (“correr”) el programa utilizando datos de entrada y resultados conocidos para verificar que se esté ejecutando el algoritmo adecuadamente ya que es posible que no existan errores de programación (sintaxis) pero los resultados finales no sean los esperados.

Implementación de algoritmos cuantitativos en Matlab.

El programador codificará su programa en el **editor de archivos** propio de Matlab mientras que la **ventana de comandos** será la interfaz entre el usuario y la máquina para la entrada/ salida de datos.



La sintaxis que se utilizará en el editor es la misma que se utiliza en la ventana de comandos. Para mayores detalles acudir al apunte **Matlab. Manejo básico**. En el siguiente punto se explican comandos básicos necesarios en la implementación de algoritmos en Matlab y que no han sido descriptos en el apunte mencionado.

Comandos básicos.

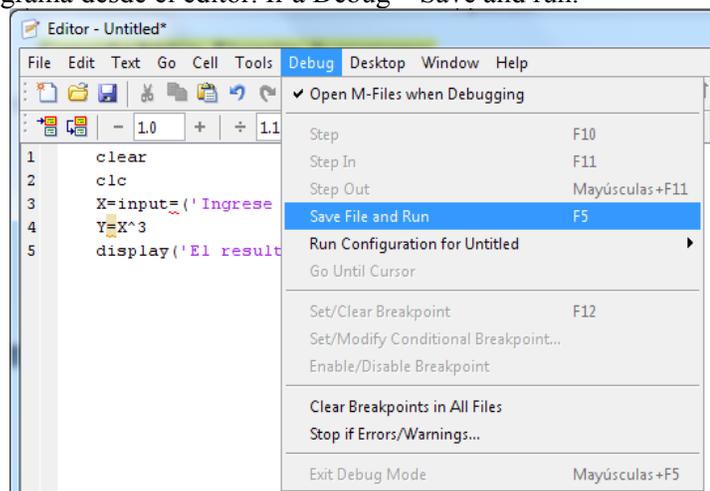
Interacción con el usuario a través de la ventana de comandos.

Ingresar valores desde la ventana de comandos	
<code>X=input('Ingrese un valor')</code>	En la ventana de comandos aparece el texto escrito entre apóstrofes. El valor ingresado por el usuario es almacenado en la variable X.
Mostrar resultados en la ventana de comandos	
<code>display('Cualquier texto')</code> <code>disp('Cualquier texto')</code>	Muestra el texto que se encuentra entre apóstrofes.
<code>display(X)</code>	Muestra el nombre de la variable (X) y el valor almacenado.
X	Escribir el nombre de la variable sin punto y coma (;) al final de la línea de comandos es equivalente a utilizar el comando display.
<code>disp(X)</code>	Muestra el valor almacenado en la variable X.
Mostrar líneas de código en la ventana de comandos	
Echo on - Echo off	Las líneas de comando que se escriban entre “echo on” y “echo off” van a ser mostradas en pantalla.

Ejecución del programa.

Para ejecutar un programa creado en el editor, se tienen diferentes alternativas:

- Ejecutar el programa desde el editor. Ir a Debug – Save and run.





- Ejecutar el programa desde el editor mediante el ícono de ejecución
- Ejecutar el programa desde el editor utilizando los atajos de teclado: presionar F5
- Ejecutar el programa desde la ventana de comandos. Deberá ingresar el nombre del archivo (inmediatamente después del prompt y sin la extensión) +enter.

Interrupción del programa.

pause	Este comando espera la respuesta del usuario. Se puede indicar la cantidad de segundos de pausa, por ejemplo pause(2) o bien el programa queda a la espera de que el usuario presione alguna tecla.
break	Este comando termina la ejecución de un bucle for o un bucle while. Siempre debe estar escrito dentro del bucle.
ctrl+c	Esta combinación de teclas se utiliza para detener la ejecución de cualquier programa de Matlab que esté activo en la ventana de comandos.

Etapas para la solución de un problema utilizando Matlab.

Mediante un ejemplo se expondrán las etapas de resolución de un problema cuantitativo utilizando Matlab.

Objetivo: Desarrollar un programa que permita calcular la tercera potencia de un valor ingresado por el usuario, mostrando el resultado final en pantalla.

- **Elaboración del algoritmo.**

Borrar variables

Limpiar pantalla

Mostrar “Ingrese el valor de la variable X”

Ingresar X

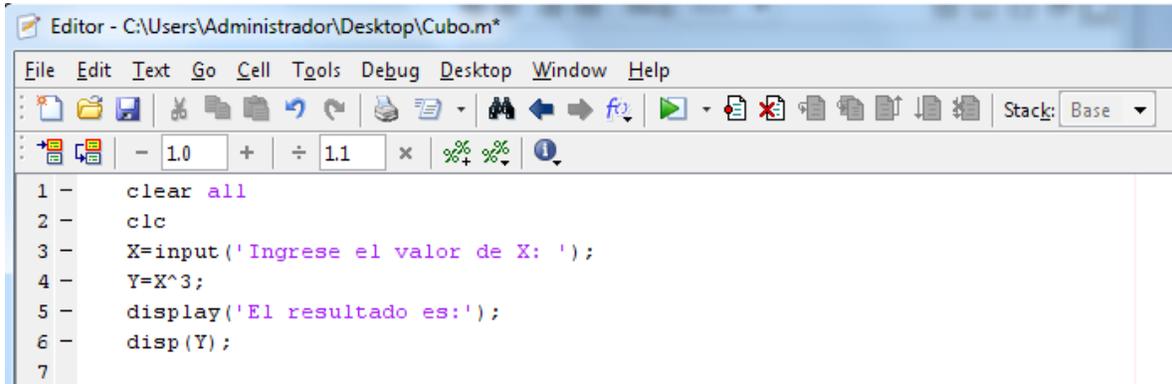
Calcular $Y=X^3$

Mostrar mensaje y resultado

- **Codificación del algoritmo en Matlab.**

El algoritmo debe ser escrito en el *editor de Matlab* de acuerdo a la sintaxis propia del programa utilizado. Se debe guardar el archivo con un nombre adecuado, llamado Cubo en este ejemplo.

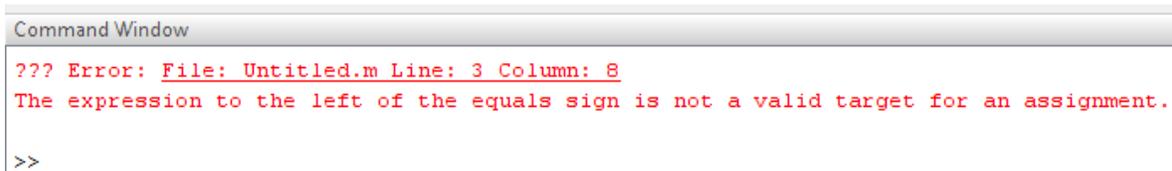
En la siguiente imagen se puede observar el código del programa implementado en el editor. En la barra de título se muestra la ruta en la que ha sido guardado el archivo de nombre Cubo.



```
Editor - C:\Users\Administrador\Desktop\Cubo.m*
File Edit Text Go Cell Tools Debug Desktop Window Help
- 1.0 + ÷ 1.1 x %>% %>
1 - clear all
2 - clc
3 - X=input('Ingrese el valor de X: ');
4 - Y=X^3;
5 - display('El resultado es:');
6 - disp(Y);
7
```

- **Ejecución. Depuración.**

Para la depuración de errores se debe “ejecutar” o “correr” el programa desde la ventana de comandos o desde el editor de Matlab. Si hay errores de sintaxis, aparecerán mensajes orientativos en la ventana de comandos (en color rojo). Por ejemplo, el siguiente mensaje informa sobre un posible error en la línea 3 – columna 8.



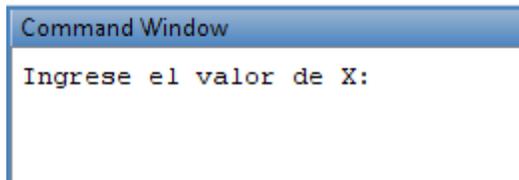
```
Command Window
??? Error: File: Untitled.m Line: 3 Column: 8
The expression to the left of the equals sign is not a valid target for an assignment.
>>
```

En general el error puede estar en esa línea o cualquier punto anterior a ella. Es importante interpretar el mensaje informado.

Si no aparecen errores, se debe pasar a la etapa de evaluación de los resultados.

- **Ejecución. Visualización y evaluación de resultados.**

Visualización en la ventana de comandos.



```
Command Window
Ingrese el valor de X:
```

Se ingresa un valor, por ejemplo 5, y se presiona “intro” así, en la ventana de comandos aparecerá el resultado:

```

Command Window
Ingrese el valor de X: 5
El resultado es:
    125
>>

```

Todas las variables definidas y sus características pueden visualizarse en el espacio de trabajo, ya que en la ventana de comandos se muestran solo los valores que se desea informar al usuario.

The screenshot shows the MATLAB Workspace window with the following table of variables:

Name	Value	Min	Max
X	5	5	5
Y	125	125	125

El programa se debe testear para varios valores de entrada, en este caso por tratarse de un algoritmo sencillo de cálculo directo no es necesario realizarlo más de una vez.

Ejemplos de aplicación.

Ejemplos con estructura secuencial pura.

1. Desarrollar un programa que pida al usuario que ingrese cinco números, se calcule la suma y el producto de dichos valores y se muestren los resultados en pantalla.

```

A=input('Ingrese un valor (A): ');
B=input('Ingrese un valor (B): ');
C=input('Ingrese un valor (C): ');
D=input('Ingrese un valor (D): ');
E=input('Ingrese un valor (E): ');

S=A+B+C+D+E;

P=A*B*C*D*E;

display('La suma de los números ingresados es igual a:')
disp(S);

display('El producto de los números ingresados es igual a: ')
disp(P);

```

2. Un sistema de ecuaciones lineales de la forma $\begin{cases} ax + by = c \\ dx + ey = f \end{cases}$ se puede resolver con las siguientes fórmulas:

$$x = \frac{ce - bf}{ae - bd} \quad y = \frac{af - cd}{ae - bd}$$

Elabore un programa que lea los coeficientes ingresados por el usuario y muestre como resultado los valores que adoptaron las incógnitas para el sistema de ecuaciones ingresado.

```
a=input('Ingrese el valor de a: ');
b=input('Ingrese el valor de b: ');
c=input('Ingrese el valor de c: ');
d=input('Ingrese el valor de d: ');
e=input('Ingrese el valor de e: ');
f=input('Ingrese el valor de f: ');
x=(c*e-b*f)/(a*e-b*d)
y=(a*f-c*d)/(a*d-b*e)
```

Ejemplos con estructuras de selección/ decisión.

3. Dado el siguiente programa, escriba el código en Matlab, ejecútelos para los siguientes valores de X: 1,8,-3,5.

```
X=input('Ingrese un número: ');

if X>0 && X<2
    Y1=X+35
elseif X>7 && X<12
    Y2=2*X^2
elseif X<0
    Y3=3*X+5
    Y4=X/100
end
```

X=1	X=8
<pre> Command Window Ingrese un número: 1 Y1 = 36 >> </pre>	<pre> Command Window Ingrese un número: 8 Y2 = 128 >> </pre>
X=-3	X=5
<pre> Command Window Ingrese un número: -3 Y3 = -4 Y4 = -0.0300 >> </pre>	<pre> Command Window Ingrese un número: 5 >> </pre>

4. Diseñe un programa que determine si un número es o no es, par positivo. Tenga en cuenta que si N es un número par:

$$(-1)^N = 1$$

```

x=input(' Ingrese un numero: ');
y=(-1)^x;
if x>0 & y==1
    disp('El número ingresado es par positivo')
else
    disp('El número ingresado NO es par positivo')
end

```

5. El precio de un boleto de viaje redondo en tren se calcula tomando en cuenta el número de kilómetros que se van a recorrer, siendo el precio \$60.00 por Km. Este precio puede tener un descuento del 30% si el viaje de regreso se hace después de 7 días del viaje de ida, o si el recorrido supera los 800 Km.

Elabore un programa que calcule cuánto se debe pagar por un boleto de tren dando como datos el total de Km recorridos y el número de días entre el viaje de ida y el viaje de vuelta.

```
km=input('Ingrese la cantidad de Km recorridos: ');
dias=input('Ingrese la cantidad de días entre el viaje de ida y
vuelta: ');

precio_km=60;

if km>800 | dias>7
    boleto=precio_km*km*0.7;
else
    boleto=precio_km*km;
end
disp('El precio de sus boletos es ($) :')
disp(boleto)
```

6. Realizar un programa que permita calcular el salario de un trabajador ingresando la cantidad de horas trabajadas y la tarifa por hora. Considere que si el trabajador ha superado las 40 horas la tarifa por hora se incrementará en un 50% para las horas extras.

```
horas=input('Ingrese las cantidad de horas trabajadas: ');
tarifa=input('Ingrese la tarifa por hora: ');

if horas<=40
    salario=tarifa*horas;
else
    horasextras=horas-40;
    salario=40*tarifa+horasextras*1.5*tarifa;
end
disp('El salario del trabajador es: ')
salario
```

7. Una agencia automotriz ofrece planes de crédito para la adquisición de los diferentes modelos de automóviles. Si el automóvil vale más de \$35,000.00 la agencia solicita un 35% de cuota inicial y el resto debe cubrirse en 24 mensualidades sin intereses. Si el automóvil tiene un valor máximo de \$35,000.00 se requiere una cuota inicial del 25% y el resto se debe pagar en 18 mensualidades sin intereses.

La agencia desea saber, en base al precio de un automóvil que un determinado cliente elija, cuál es el monto de la cuota inicial; y cuál es el número y el monto de las mensualidades que éste debe cubrir.

```
precio=input('Ingrese el precio del automóvil: ');  
  
if precio>35000  
    cant_cuotas=24;  
    cuota_inicial=precio*0.35;  
else  
    cant_cuotas=18;  
    cuota_inicial=precio*0.25;  
end  
  
cuota_mensual=(precio-cuota_inicial)/cant_cuotas;  
  
disp('Cuota inicial ($)')  
disp(cuota_inicial)  
disp('Cantidad de cuotas :')  
disp(cant_cuotas)  
disp('Cuota mensual ($) :')  
disp(cuota_mensual)
```

8. Implemente un algoritmo determine el máximo de tres números ingresados por el usuario. Utilizar expresiones condicionales. Se asume que los números ingresados son diferentes.

```
x=input(' Ingrese el primer número: ');  
y=input(' Ingrese el segundo número: ');  
z=input(' Ingrese el tercer número: ');  
  
if x>y & x>z
```

```

disp('El máximo es: ')
display(x)
elseif y>x & x>z
disp('El máximo es: ')
display(y)
elseif z>x & z>y
disp('El máximo es: ')
display(z)
end

```

Realice el ejercicio almacenando los valores ingresados en un vector y utilice la función de Matlab que permite buscar el máximo número en un vector.

```

x=input(' Ingrese tres valores como elementos de un vector: ') ;
y=max(x);
disp('El mayor es: ');
disp(y);

```

Como modificaría el algoritmo si además del valor máximo desea conocer su posición en el vector?

```

x=input(' Ingrese tres valores como elementos de un vector: ') ;
[y,c]=max(x);
disp('El mayor valor es: ');
disp(y);
disp('Ubicado en la posición:')
disp(c);

```

9. Elabore un programa que permita calcular las raíces reales de un polinomio de segundo grado de la forma $ax^2 + bx + c = 0$. Tenga en cuenta que dependiendo del valor del determinante ($D=b^2-4ac$) el sistema puede dar raíces reales ($D \geq 0$) o raíces complejas ($D < 0$). El programa debe informar al usuario si el sistema no tiene raíces reales. Además se debe indicar si las raíces reales son iguales ($D=0$) o diferentes ($D > 0$).

```
a=input('Ingrese el valor de a: ');
b=input('Ingrese el valor de b: ');
c=input('Ingrese el valor de c: ');

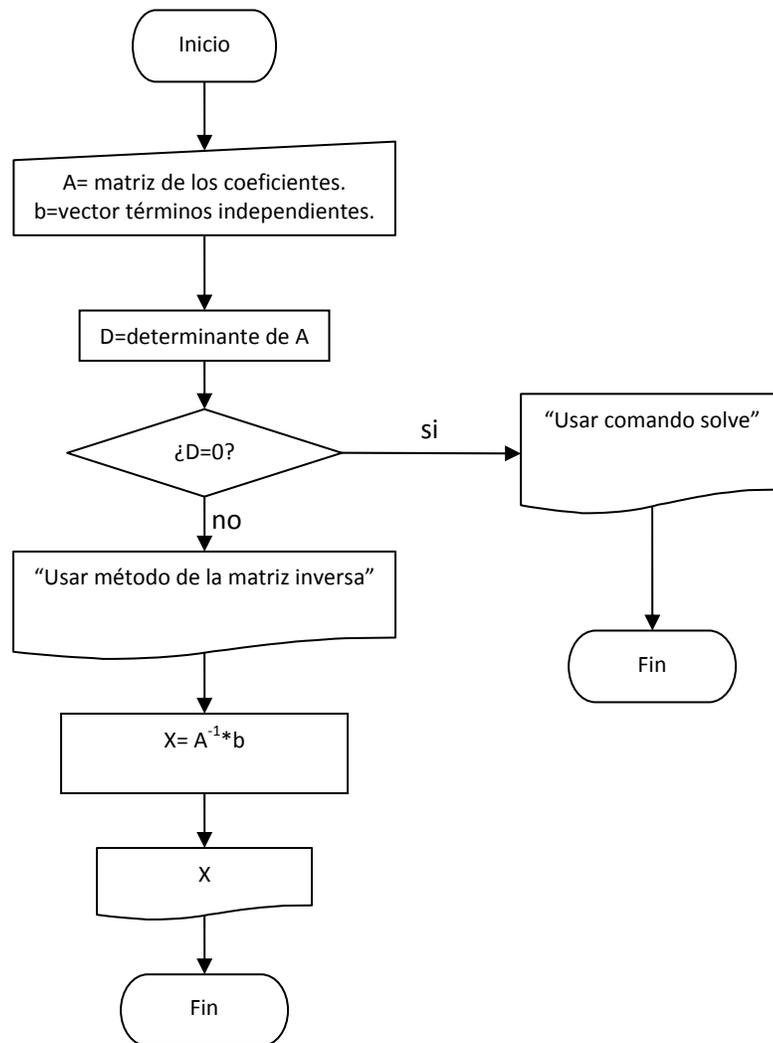
D=b^2-4*a*c;

if D<0
    display('El polinomio no tiene raíces reales')
elseif D==0

    display('Raíces reales iguales')
    x1=(-b+D^0.5)/(2*a);
    disp(x1)
else

    x1=(-b+D^0.5)/(2*a);
    x2=(-b-D^0.5)/(2*a);
    display('Raíces reales diferentes')
    display(x1)
    display(x2)
end
```

10. El siguiente diagrama de flujo representa un algoritmo de cálculo para resolver un sistema de ecuaciones. Implemente el algoritmo en Matlab.



```
A=input('Ingrese la matriz de coeficientes A');  
b=input('Ingrese el vector de términos independientes b');  
D=det(A);  
if D==0;  
    disp('Usar el comando solve')  
else  
    disp('Usar el metodo de la matriz inversa')  
    X=inv(A)*b;  
    disp('las soluciones son')
```

```
x
end
```

11. Desarrolle un programa que permita determinar el valor de la función (y) para un valor de x ingresado por el usuario.

$$y = f(x) = \begin{cases} 3x+36 & \text{si } x \leq 11 \\ x^2 - 10 & \text{si } 11 < x \leq 33 \\ x+6 & \text{si } 33 < x \leq 64 \\ 0 & \text{para los demás valores de } x \end{cases}$$

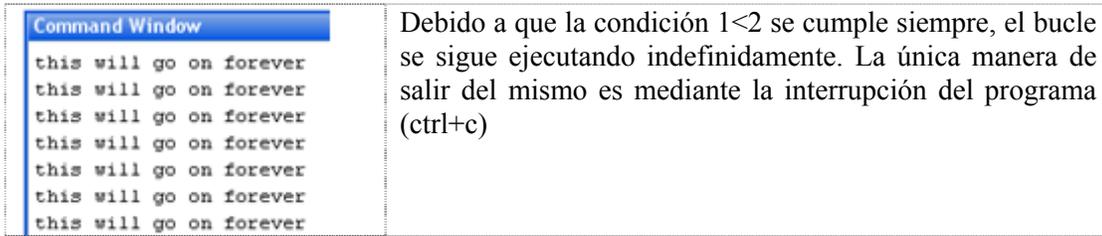
```
x=input('Ingrese un valor: ')

if x<=11
    y=3*x+36;
elseif x<=33
    y=x^2-10;
elseif x<=64
    y=x+6;
else
    y=0;
end
display(y)
```

Ejemplos con estructuras de repetición.

12. Copie, pegue y ejecute el siguiente código. Explique qué es lo que sucede.

```
while 1<2
disp('this will go on forever')
end
```



13. Hacer un algoritmo que pida al usuario un número N e imprima números sucesivos (1,2,3,4...) hasta que el último sea el anterior a N.

- Utilice la estructura while-end
- Utilice la estructura for-end

Usando la estructura while-end
<pre> N=input('Ingrese un valor entero positivo: '); i=1; while i<N i i=i+1; end </pre>
Usando la estructura for-end
<pre> N=input('Ingrese un valor entero positivo: '); for i=1:N-1 i end </pre>

14. Realice un programa que realice la suma de todos los números enteros pares comprendidos entre 1 y 100 ($1 \leq N \leq 100$).

- Utilice la estructura while-end
- Utilice la estructura for-end

Usando la estructura while-end
<pre> a=2; suma=0; while a<=100; </pre>

```
suma=suma+a;  
a=a+2;  
end  
disp('La suma de todos los números pares entre 1 y 100 es: ')  
disp(suma)
```

Usando la estructura for-end

```
suma=0;  
for i=2:2:100  
    suma=suma+i;  
end  
disp('La suma de todos los números pares entre 1 y 100 es: ')  
disp(suma)
```

15. Desarrollar un programa que a partir de un número N ingresado por el usuario, muestre números sucesivos menores que N. Asuma que se ingresa un número entero positivo.

- a) Utilice la estructura while-end
- b) Utilice la estructura for-end

Usando la estructura while-end

```
N=input('Ingrese un valor entero positivo: ');  
i=N;  
while i<=N && i>=1  
i  
i=i-1;  
end
```

Usando la estructura for-end

```
N=input('Ingrese un valor entero positivo: ');  
for i=N:-1:1  
i  
end
```

16. Desarrolle un programa que pida al usuario que ingrese un número mientras el número sea distinto de “7”. Muestre en pantalla la cantidad de veces que se ha ingresado un número.

```
L=0;

N=0;

while N~=7

N=input('Ingrese un número: ');

L=L+1;

end

display('Cantidad de intentos: ');

disp(L);
```

17. Desarrolle un programa que calcule el duplo de un número (n) con $1 \leq n \leq 5$ y almacene cada uno de los resultados.

- a) Utilice la estructura for-end
- b) Utilice la estructura while-end

a) Utilizando la estructura for-end	b) Utilizando la estructura while-end
<pre>for n=1:5 x(n)=2*n; end x</pre>	<pre>n=1; while n<6 x(n)=2*n; n=n+1; end x</pre>

18. Elabore un programa que calcule el área total y el volumen del cilindro ingresando el radio y la altura de un cilindro. Considerar que es erróneo el ingreso de valores negativos de altura y radio, ya que no puede obtenerse como resultado un valor de volumen o área negativo.

Tenga en cuenta que el área de un cilindro es igual a la suma de las áreas de los dos círculos más el área del rectángulo formado por la longitud de la circunferencia y la altura del cilindro y que el volumen se obtiene multiplicando la superficie de la base del círculo por la altura del cilindro.

```
R=input('ingrese el radio del cilindro (cm): ');  
  
while R<0  
R=input('ingrese el radio del cilindro (cm): ');  
end  
  
H=input('ingrese la altura del cilindro (cm): ');  
  
while H<0  
H=input('ingrese la altura del cilindro (cm): ');  
end  
  
Area=2*pi*R^2+2*pi*R*H;  
Volumen= pi*R^2*H;  
  
display('El área calculada es (cm2): ')  
disp(Area)  
display('El volumen calculado es (cm3): ')  
disp(Volumen)
```

19. Diseñe un programa que pida por teclado una temperatura en °C, la convierta a °K y muestre el resultado. Considere que el usuario requiere realizar la conversión un número indefinido de veces.

```
m=1;  
  
while m==1  
t=input('Ingrese la temperatura en °C: ');  
display('Temperatura (K): ')  
T=t+273.15;  
disp(T)  
  
m=input('Presione 1 para continuar o cualquier otra tecla para  
terminar');  
  
end
```

Que cambios realizaría en el programa si desea almacenar todos los valores que ha convertido?

```
m=1;

i=0;

while m==1

    i=i+1;

    t=input('Ingrese la temperatura en °C: ');

    T(i)=t+273.15;

    m=input('Presione 1 para continuar o cualquier otra tecla para
terminar');

end

display('Temperatura (K): ')

disp(T)
```

La temperatura calculada (K) se guarda en la posición i del vector T.

Que cambios realizaría en el programa si desea almacenar en una sola tabla los valores ingresados en °C y los valores convertidos a °K?

```
m=1;

i=0;

while m==1

    i=i+1;

    t(i,1)=input('Ingrese la temperatura en °C: ');

    t(i,2)=t(i,1)+273.15;

    m=input('Presione 1 para continuar o cualquier otra tecla para
terminar');

end

display('Temp. (C) Temp. (K): ')
```

```
disp(t)
```

La temperatura ingresada por el usuario se almacena en la fila i , columna 1 de la matriz t , mientras que la temperatura calculada (K) se almacena en la fila i , columna 2 de la matriz t .

20. Realice un programa que genere un vector u , constituido por los números enteros pares del vector v , donde $v=1:1:50$. Utilice la estructura for-end

```
v=1:1:50;
for i=2:2:50
    u(i/2)=v(i);
end
disp('Los números pares son:')
disp(u')
```

21. Desarrolle un programa que permita calcular el área y perímetro de círculos y los grafique en función del radio para todos los valores ingresados. Considere que no conoce la cantidad de datos que ingresará el usuario.

- Utilice la estructura for-end
- Utilice la estructura while-end

Usando la estructura while-end

```
m=1;
i=1;
while m==1
    r(i)=input('Ingrese el radio: ');
    Area(i)=pi*r(i)^2;
    Perimetro(i)=2*pi*r(i);
    m=input('Ingrese 1 para continuar, cualquier tecla para finalizar');
    i=i+1;
end
L=length(r);
```

```

if L>0

plot(r,Area,'r*-',r,Perimetro,'b. ');

legend('Area','Perímetro')

end

```

Usando la estructura for-end

```

r=input('Ingrese los valores de radio en forma vectorial: ');

L=length(r);

for i=1:r

Area=pi*r.^2;

Perimetro=2*pi.*r;

end

if L>0

plot(r,Area,'r*-',r,Perimetro,'b. ');

legend('Area','Perímetro')

end

```

22. El peso de una muestra se determina realizando 4 pesadas, se descarta el menor peso y se promedian los otros tres. Considere que el usuario ingresa una tabla con el número de sustancia en la primera columna y las mediciones realizadas en las otras 4 columnas. Diseñe un programa que muestre en forma matricial, el número de muestra, las mediciones, el peso eliminado y el peso promedio de cada sustancia pesada.

```

A=input('Ingrese la tabla de datos: ');

[f,c]=size(A);

for i=1:f

v=A(i,2:c);

minimo(i)=min(v);

promedio(i)=(sum(A(i,2:c))-minimo(i))/3;

end

```

```

A(:,c+1)=minimo;
A(:,c+2)=promedio;
disp('ID Muestra      Peso1      Peso2      Peso3      Peso4      Min.
Prom. ')
disp(A)

```

Tomando como datos de entrada la siguiente tabla,

ID Muestra	Peso 1	Peso 2	Peso 3	Peso 4
1	5.13	6.12	6.10	5.95
2	4.68	4.65	4.39	4.75
3	6.42	6.38	6.3	6.43

Los resultados son:

```

Command Window
Ingrese la tabla de datos: [
  1 5.13 6.12 6.10 5.95
  2 4.68 4.65 4.39 4.75
  3 6.42 6.38 6.3  6.43
];
ID Muestra      Peso1      Peso2      Peso3      Peso4      Min.      Prom.
  1.0000      5.1300      6.1200      6.1000      5.9500      5.1300      6.0567
  2.0000      4.6800      4.6500      4.3900      4.7500      4.3900      4.6933
  3.0000      6.4200      6.3800      6.3000      6.4300      6.3000      6.4100
>>

```

Ejemplos con estructuras anidadas.

23. Copie, pegue y ejecute el siguiente código. Explique qué es lo que sucede.

```

while 1
disp('¿Qué es 3+4?')
respuesta=input('');
    if respuesta == 7
disp('Correcto')
    else

```

<pre> disp('Erróneo, pero te daré otra oportunidad.') end end </pre>	
<pre> Command Window ¿Qué es 3+4? 7 Correcto ¿Qué es 3+4? 9 Erróneo, pero te daré otra oportunidad. ¿Qué es 3+4? 7 Correcto ¿Qué es 3+4? 7 Correcto ¿Qué es 3+4? 11 Erróneo, pero te daré otra oportunidad. ¿Qué es 3+4? </pre>	<p>El bucle se sigue ejecutando indefinidamente, independientemente de que se ingrese el número correcto (7) ya que no hay establecida una condición en la sentencia de repetición (while). La única manera de salir del mismo es mediante la interrupción del programa (ctrl+c)</p>

24. Desarrolle un programa que sume los números almacenados en un vector hasta que encuentre el número cero. Por ejemplo, para el vector $v=[1\ 4\ 3\ 0\ 5\ 1]$ el resultado será $1+4+3=8$.

- Utilice la estructura for-end
- Utilice la estructura while-end

<i>Usando la estructura for-end</i>
<pre> v=input('Ingrese los elementos del vector: '); suma=0; n=length(v); for i=1:n if v(i)==0; break end suma=suma+v(i); end suma </pre>
<i>Usando la estructura while-end</i>
<pre> v=input('Ingrese los elementos del vector: '); </pre>

```

i=1;
suma=0;
n=length(v);

while v(i)~=0
    suma=suma+v(i);
    i=i+1;
    if i==n+1;
        break
    end
end

suma

```

Observación: En el caso a) el comando **break** se usa para detener la suma si aparece un elemento con valor cero, mientras que en el caso b) se agrega la restricción para detener la suma en el caso que todos los elementos sean diferentes de cero. Verifique la función de este comando quitando esta restricción para el caso en el que el vector sea por ejemplo [1,1,1,1].

25. Realice un programa que genere un vector u , constituido por los números enteros pares de un vector v , ingresado por el usuario. El programa deberá imprimir un vector u y la suma de todos los elementos. Se asume que el usuario ingresará solo números enteros.

Pruebe el programa ingresando el vector: $v = [9,6,4,4,2,-4,3,2,-90,1,1,3,6,5]$

```

v=input('Ingrese un vector de números enteros: ');
n=length(v);
j=1;
for i=1:n;
    s= (-1)^v(i);
    if s==1
        u(j)=v(i);
        j=j+1;
    end
end

```

```

suma=sum(u);

disp('Los números pares son:')

disp(u')

disp('La suma de todos los elem. pares del vector v es: ')

disp(suma)

```

Ejecutando el programa para el vector dado, los resultados son:

```

Command Window
Ingrese un vector de números enteros: [9,6,4,4,2,-4,3,2,-90,1,1,3,6,5]
Los números pares son:
    6
    4
    4
    2
   -4
    2
  -90
    6

La suma de todos los elem. pares del vector v es:
   -70

```

26. Desarrolle un programa que le permita determinar de una lista de números de tamaño indeterminado:

- Cuántos valores son mayores o iguales a 50 y menores o iguales a 75 ($50 \leq x \leq 75$)
- Cuántos valores son mayores a 80 ($x > 80$)
- Cuántos valores son menores a 30 ($x < 30$)

```

x=input('Ingrese los valores en forma vectorial :')
N=length(x);
cuenta1=0;
cuenta2=0;
cuenta3=0;

for i=1:N
    if x(i)>=50 & x(i)<=75
        cuenta1=cuenta1+1;
    end
end

```

```

elseif x(i)>80
    cuenta2=cuenta2+1;
elseif x(i)<30
    cuenta3=cuenta3+1;
end
end

disp('La cantidad de valores entre 50 y 75 son:')
disp(cuenta1)
disp('La cantidad de valores mayores a 80 son:')
disp(cuenta2)
disp('La cantidad de valores menores a 30 son:')
disp(cuenta3)

```

27. Desarrolle un programa que le permita determinar de una matriz ingresada por el usuario (tamaño desconocido):

- Cuantos valores son mayores o iguales a 5 y menores o iguales a 7 ($5 \leq x \leq 7$)
- Cuantos valores son mayores a 8 ($x > 8$)
- Cuantos valores son negativos ($x < 0$)

```

x=input('Ingrese los valores en forma matricial :')
[F,C]=size(x);
cuenta1=0;
cuenta2=0;
cuenta3=0;

for i=1:F
    for j=1:C
        if x(i,j)>=5 & x(i,j)<=7
            cuenta1=cuenta1+1;
        elseif x(i,j)>8

```

```
        cuenta2=cuenta2+1;
    elseif x(i,j)<0
        cuenta3=cuenta3+1;
    end
end
end

disp('La cantidad de valores entre 5 y 7 son:')
disp(cuenta1)
disp('La cantidad de valores mayores a 8 son:')
disp(cuenta2)
disp('La cantidad de valores negativos son:')
disp(cuenta3)
```

28. Dada una matriz B ingresada por el usuario de tamaño nxm:

Se pide evaluar cada elemento de B con respecto a las siguientes condiciones:

- Si el valor $B_{i,j}$ es múltiplo de 2, entonces se deberá elevar al cuadrado ese valor y almacenarlo en la matriz Y en la posición i,j .
- Si el valor $B_{i,j}$ es múltiplo de 3, entonces se deberá multiplicar ese valor por 2 y almacenarlo en la matriz Y en la posición i,j .
- En caso de que no se de ninguna de las condiciones mencionadas, el valor $B_{i,j}$ se deberá igualar a 0 y almacenar el resultado en la matriz Y en la posición i,j .

En la ventana de comandos solo deberán mostrarse la matriz B y la matriz Y.

Nota: El comando “rem” se utiliza para calcular el resto de una división por lo que, me permite saber si un número es múltiplo de otro. Ver “help rem”.

```
B=[1,2,3;4,5,6;7,8,9];

[f,c]=size(B);

for i=1:f
```

```

for j=1:c
    if rem(B(i,j),2)==0
        Y(i,j)=B(i,j)^2;
    elseif rem(B(i,j),3)==0
        Y(i,j)=B(i,j)*2;
    else
        Y(i,j)=0;
    end
end
end

B
Y

```

29. La ecuación de estado de Peng – Robinson es:

$$P = \frac{RT}{v-b} - \frac{a\alpha}{v(v+b)+b(v-b)}$$

Esta ecuación puede reescribirse como un polinomio de tercer grado:

$$z^3 - (1-B)z^2 + (A-3B^2-2B)z - (AB-B^2-B^3) = 0$$

$$A = \frac{aP}{R^2T^2} \quad B = \frac{bP}{RT}$$

$$a = 0.45724 \frac{R^2T_c^2}{P_c} \alpha \quad b = 0.07780 \frac{RT_c}{P_c}$$

$$\alpha = \sqrt{1 + \kappa \left(1 - \left(\frac{T}{T_c} \right)^{0.5} \right)} \quad \kappa = 0.37464 + 1.54226 \omega - 0.26992 \omega^2$$

Donde la presión está expresadas en atm, la temperatura en °K y la constante universal de los gases ideales en at.l/°K mol.

Implemente un algoritmo en Matlab que calcule el valor del factor de compresibilidad z (es el valor máximo de las 3 raíces del polinomio – busque en la ayuda de Matlab el comando para encontrar el máximo de un conjunto de valores) y a partir de éste calcule el volumen específico

del CO₂ para una presión y temperatura ingresados por el usuario. Incluya condiciones que aseguren presiones positivas y temperaturas mayores a 273.15 K.

$$v = \frac{zRT}{P}$$

Tc (°C)	31.10
Pc (bar)	7.383e6
ω	0.228

Aclaración: hacer el pasaje de unidades correspondiente.

```
T=200;

while T<273.15;
T=input('Ingrese la temperatura del gas [K]: ');
    if T<273.15
        disp('El valor ingresado como temperatura no es válido');
        disp('La temp. debe ser mayor o igual a 273.15 K');
    end

end

P=-5;

while P<=0
    P=input('Ingrese la presión del gas [atm]: ');
    if P<=0
        disp('El valor ingresado como presión no es válido');
        disp('La presión debe ser mayor a 0 atm');
    end

end

R=0.082;
```

```

Tc=31.10+273.15;
Pc=7.383e6*0.986923;
w= 0.228;

k=0.37464+1.54226*w-0.26992*w^2;

alfa=(1+k*(1-(T/Tc)^0.5))^0.5;

a=0.45724*alfa*(R*Tc)^2/Pc;

b=0.07780*R*Tc/Pc;

A=a*P/(R*T)^2;

B=b*P/(R*T);

raices=roots([1,-(1-B),(A-3*B^2-2*B),-(A*B-B^2-B^3)]);
z=max(raices);

disp('El vol. esp. del gas a las condiciones ingresadas es: ')
v=R*T*z/P

```

30. Desarrolle un algoritmo que permita calcular el calor específico de las sustancias en función de la temperatura de acuerdo a la siguiente expresión:

$$C_{p_i} = a_i + b_i T + c_i T^2 \quad \text{donde } C_p \text{ está en [J/mol K] y la temperatura en [K]}$$

El programa debe seguir la siguiente estructura:

- El usuario debe ingresar:
 - La cantidad de sustancias (n) de las que calculará la capacidad calorífica.

- Los coeficientes de cada sustancia (a, b, c).
- La temperatura (T) en K teniendo en cuenta que debe ser mayor o igual a 298.15 K.
- Almacenar los coeficientes en la matriz (datos) de tamaño nx3.
- Calcular la capacidad calorífica y almacenar los resultados en un vector (cp).
- Mostrar los resultados en pantalla.

A modo de ejemplo y comparación de resultados, en la siguiente tabla se muestran los valores de los coeficientes para algunas sustancias y el calor específico calculado a 25°C.

Probar si el programa funciona calculando el calor específico:

- Para un sola sustancia
- Para dos sustancias
- Para cuatro sustancias

Gas	<i>a</i>	<i>b</i>	<i>c</i>	Cp (298.15)
CO	26.16	8.75e-3	-1.92e-6	28.6
O ₂	25.66	1.25e-2	-3.37e-6	29.9
CO ₂	28.67	3.57e-2	-1.07e-5	38.37
N ₂	26.37	7.61e-3	-1.44e-6	28.51

```
n=input('Ingrese la cantidad de sustancias: ');

for j=1:n
    datos(j,1)=input('Ingrese a: ');
    datos(j,2)=input('Ingrese b: ');
    datos(j,3)=input('Ingrese c: ');
    if n>1 && j~=n
        disp('Continuar con la sustancia siguiente')
    end
end
```

```

T=200;

while T<298.15
    T=input('Ingrese la temperatura (K): ');
    if T<298.15
        disp('La temp. debe ser mayor o igual a 298.15 K');
    end
end

for i=1:n
    cp(i)=(datos(i,1)+datos(i,2)*T+datos(i,3)*T^2)';
end

disp('Capacidad calorífica [J/mol K]: ')
disp(cp)

```

31. Se desea analizar la velocidad de variación de la temperatura de una reacción exotérmica irreversible con el tiempo de reacción. Diseñe un programa que pida por teclado el tiempo y la temperatura en °C. Almacene todos los datos ingresados. Transforme la temperatura de °C a °K y grafique la variación de la temperatura con el tiempo. Considere que se ingresarán los datos hasta que la variación de la temperatura con el tiempo entre mediciones sea inferior a 0.2°K.

```

dif_T=0.5;

i=1;

while dif_T>0.2

    x(i)=input('ingrese el tiempo');
    t(i)=input('ingrese la temp. ');

    if i==1
    else

```

```

    dif_T=t(i)-t(i-1);

    end

    i=i+1;

end

T=t+273.15;

plot(x,T,'g*-');

title('Variación de la temp. de reacción con el tiempo')

xlabel('Tiempo (seg)')

ylabel('Temperatura (K)')

```

32. Se desea desarrollar un programa que permita graficar la pérdida de carga que se produce debido al flujo de un fluido dentro de una cañería en función del número de Reynolds.

La caída de presión se determina a partir de las siguientes ecuaciones:

$$\Delta P = 4 \cdot f \cdot \frac{L}{D_i} \cdot \frac{G_t^2}{2 \cdot \rho \cdot 1000}$$

Donde:

ΔP : pérdida de carga (kPa)

f: factor de fricción de Fanning (adimensional)

L: longitud de la cañería (m)

D_i : diámetro interno de la cañería (m)

G_t : velocidad másica de flujo por unidad de área (Kg/m² seg)

ρ : densidad del fluido (Kg/m³)

A su vez, el factor de fricción “f” depende del Número de Reynolds, el cual se expresa de la siguiente manera:

$$Re = \frac{D_i \cdot G_t}{\mu}$$

donde μ es la viscosidad del fluido en (Kg/m seg) y el n° de Re es adimensional.

La expresión que calcula el factor de fricción para $Re \leq 2100$ es,

$$f = \frac{16}{Re}$$

y para $Re > 2100$ se utiliza,

$$f = 0,0014 + 0,125 \cdot Re^{-0,32}$$

Considere que el usuario puede ingresar las dimensiones de la cañería, las características del fluido y el rango de velocidades másicas en el que desea evaluar la caída de presión.

```
Gt=input('Ingrese los valores de velocidad másica [(Kg/m^2 seg)]: ');
n=length(Gt);
disp('Dimensiones de la cañería')
Di=input('Ingrese el diámetro de la cañería [m]: ');
L=input('Ingrese la longitud de la cañería [m]: ');
disp('Propiedades del fluido: ');
rho=input('Ingrese la densidad [Kg/m^3]: ');
mu=input('Ingrese la viscosidad [Kg/m seg]: ');

Re=Di.*Gt/mu;

j=1;
k=1;
for i=1:n
    if Re(i)>2100
```

```

        f=0.0014+0.125*Re(i)^(-0.32);
        dp1(j)=4*f*L*Gt(i)^2/(Di*rho*2*1000);
        re1(j)=Re(i);
        j=j+1;
    else
        f=16/Re(i);
        dp2(k)=4*f*L*Gt(i)^2/(Di*rho*2*1000);
        re2(k)=Re(i);
        k=k+1;
    end
end

plot(re1,dp1,'-b',re2,dp2,'-r')
title('Pérdida de carga en función del número de Reynolds')
legend('Re>2100','Re<=2100')
xlabel('Re')
ylabel('dp [kPa]')

```

Si el fluido que circula por una cañería de 1 pulg (0.0254 m) y 25 m de longitud es agua a 25 C ($\mu=1e-3$ Kg/m seg; $\rho=1000$ Kg/m³). La gráfica de la pérdida de carga en función del número de Re para $1 \leq Gt \leq 200$ será:

```

Command Window
Ingrese los valores de velocidad másica [(Kg/m^2 seg)]: linspace(1,200,150)
Dimensiones de la cañería
Ingrese el diámetro de la cañería [m]: 0.0254
Ingrese la longitud de la cañería [m]: 25
Propiedades del fluido:
Ingrese la densidad [Kg/m^3]: 1000
Ingrese la viscosidad [Kg/m seg]: 1e-3
>>

```

