

# **PROGRAMACION I - PRÁCTICA**

- **INTRODUCCIÓN A LA PROGRAMACIÓN**
- **DEFINICIONES, INSTRUCCIONES BÁSICAS, VARIABLES Y OPERADORES**

# DEFINICIONES BÁSICAS

- El objetivo de la materia es **la introducción a los conocimientos básicos de la programación de computadoras.**
- Programar es hacer programas, utilizando para ello algún **lenguaje.**
- Un **lenguaje de programación** nos proporciona un conjunto de instrucciones, mecanismos y reglas, mediante las cuales construir el código capaz de resolver un problema.
- En PROGRAMACION I, parte práctica, vamos a aprender a programar con el **lenguaje C/C++**, en el entorno de desarrollo **Code::Blocks.**

# DEFINICIONES BÁSICAS

- **Programa de computación:** conjunto o serie de instrucciones que permiten arribar a un resultado o a la solución de un problema.
- Las **instrucciones** están basadas en reglas establecidas, comprensibles para la computadora que se pretende utilizar, y su ejecución debe producirse de manera ordenada (una secuencia única que no se puede alterar).
- Estas instrucciones, así como las reglas para su escritura **son específicas de cada lenguaje de programación**. No obstante lo anterior, todos los lenguajes se basan en un conjunto muy limitado de estructuras que veremos a lo largo del curso.
- **Como nuestro objetivo es programar, vamos a aprender a analizar cuidadosamente el problema a resolver, para dividirlo en las partes más pequeñas que lo constituyen para luego resolverlo con las instrucciones que correspondan.**

# COMPONENTES DIAGRAMA FLUJO

- Ingreso de datos



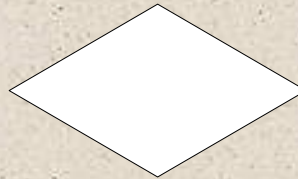
- Salida de datos



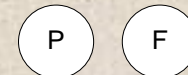
- Operación



- Decisión simple



- Indicadores de principio y fin del programa



# PROGRAMA EJEMPLO.

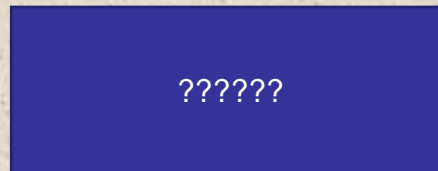
## ANÁLISIS DEL PROBLEMA

- Dados dos números que se ingresan por teclado, calcular e informar la suma de ambos.

Datos de entrada



Acciones para transformar  
entrada en salida pedida



Salida



# **PROGRAMA EJEMPLO.**

## **ANÁLISIS DEL PROBLEMA**

- Dados dos números que se ingresan por teclado, calcular e informar la suma de ambos.
- Datos de entrada: 2 números
- Salida: 1 número
- Acciones para transformar los datos de entrada en la salida pedida: operación de suma de ambos números

# PROGRAMA EJEMPLO (instrucciones)

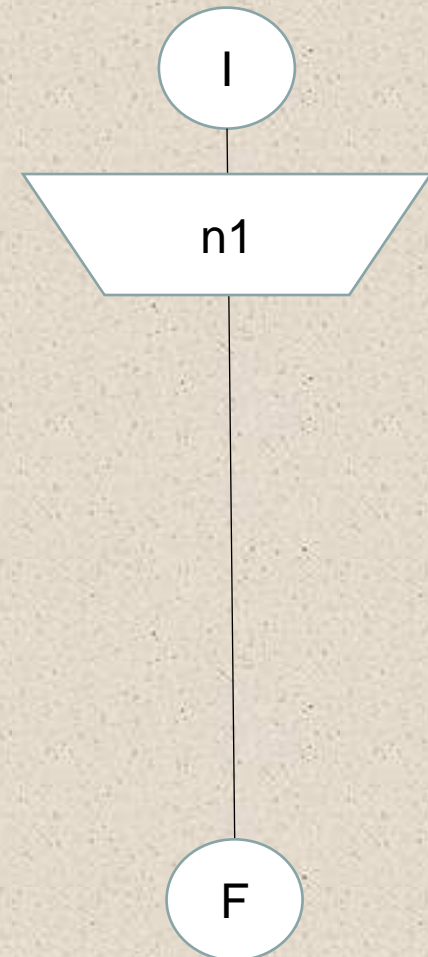
- Ingresar el primer número (entrada)
- Ingresar el segundo número (entrada)
- Calcular la suma (operación)
- Mostrar el resultado la suma (salida)

# PROGRAMA EJEMPLO

## (diagrama y código)

Dados dos números, calcular e informar la suma de ambos

- Ingresar el primer número (entrada)  
`cin>>n1;`



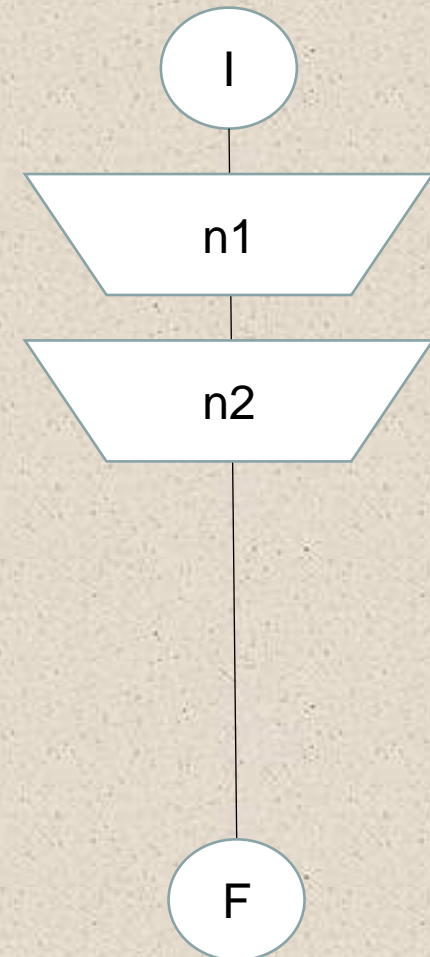


# PROGRAMA EJEMPLO

## (diagrama y código)

Dados dos números, calcular e informar la suma de ambos

- Ingresar el primer número (entrada)  
`cin>>n1;`
- Ingresar el segundo número (entrada)  
`cin>>n2;`

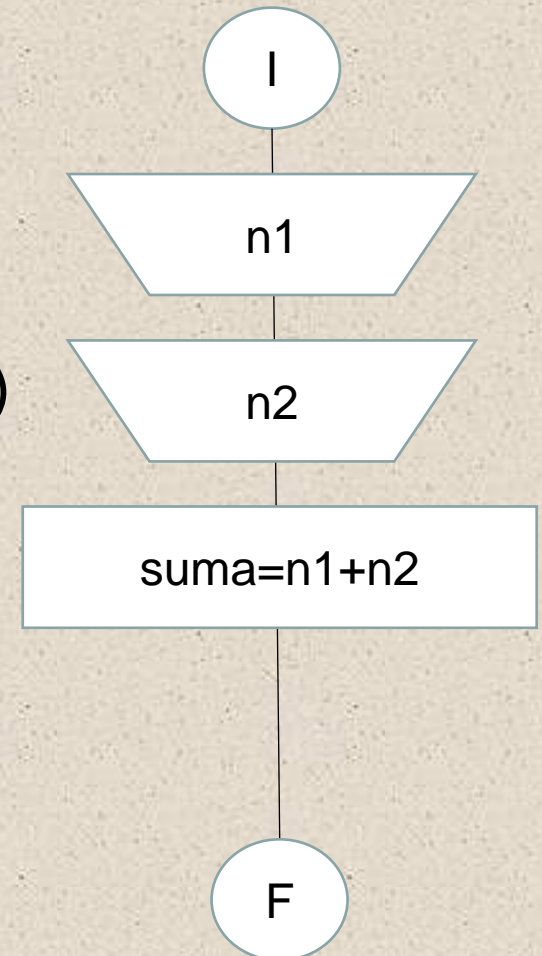


# PROGRAMA EJEMPLO

## (diagrama y código)

Dados dos números, calcular e informar la suma de ambos

- Ingresar el primer número (entrada)  
`cin>>n1;`
- Ingresar el segundo número (entrada)  
`cin>>n2;`
- Calcular la suma (operación)  
`suma=n1+n2;`

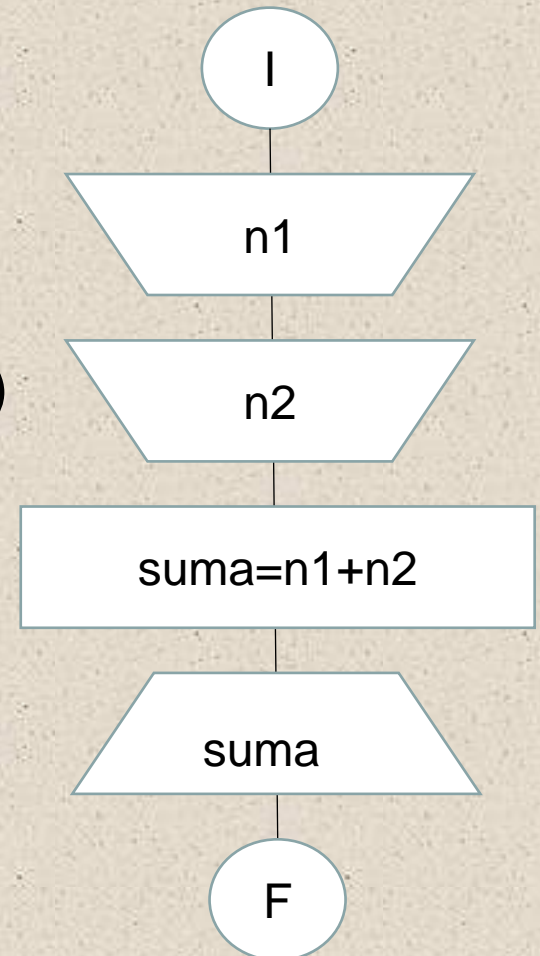


# PROGRAMA EJEMPLO

## (diagrama y código)

Dados dos números, calcular e informar la suma de ambos

- Ingresar el primer número (entrada)  
`cin>>n1;`
- Ingresar el segundo número (entrada)  
`cin>>n2;`
- Calcular la suma (operación)  
`suma=n1+n2;`
- Mostrar el resultado de la suma (salida)  
`cout<<suma;`



# PROGRAMA EJEMPLO

## (código lenguaje C++)

Dados dos números, calcular e informar la suma

```
# include<iostream>
# include<cstdlib>

using namespace std;

int main(){
    int n1,n2, suma;           //DECLARACION DE VARIABLES
    cin>>n1;                   // INGRESO DE DATOS
    cin>> n2;                   // INGRESO DE DATOS
    suma=n1+n2;                // OPERACION CALCULO DE LA SUMA
    cout<<suma;                //SALIDA POR PANTALLA
    return 0;
}
```

# DE CODIGO FUENTE A PROGRAMA EJECUTABLE

Código fuente



Acciones para transformar  
fuente en ejecutable

COMPILADOR  
C/C++



Programa ejecutable



# DE CODIGO FUENTE A PROGRAMA EJECUTABLE

- **Código fuente:** archivo de texto que escribimos, respetando estrictamente las reglas que el lenguaje de usemos nos indique. Nosotros entendemos las instrucciones que el archivo contiene
- **Compilación:** traducción del contenido del archivo fuente en instrucciones que sean comprensibles para la máquina.
- **Programa o archivo ejecutable:** conjunto de instrucciones que la máquina entiende y puede ejecutar.

# ELEMENTOS DEL LENGUAJE

Cada una de las palabras y los símbolos que se utilizan en los programas deben ser los que el lenguaje a utilizar acepte o exija.

Los elementos básicos son:

- Variables
- Constantes
- Operadores
- Expresiones

# VARIABLES

- Definición

Representación simbólica de espacio de memoria. Es donde se almacenan los datos en procesamiento.

Una variable se identifica con un nombre (lo elige el programador), y permite escribir un dato en la memoria o leer un dato de la memoria

- Clasificación:

- Por tipo de dato que almacena (p.e. números)

- Por alcance (que parte del programa lo conoce)

- Por la cantidad de elementos que puede almacenar

# VARIABLES

- Declaración de variables

tipo\_de\_dato            identificador

- Ejemplo en C/C++  
    int dato1;  
    float dato2;

# CONSTANTES

- Definición:  
Espacio de memoria que permite almacenar un dato que no puede modificarse.

Una constante se identifica con un nombre (lo elige el programador), y el dato que contiene se le asigna al declararla

- Declaración de constantes  
`const tipo_de_dato IDENTIFICADOR=VALOR`



# OPERADORES BASICOS

- **Asignación** =
- **Matemáticos**

	<b>+</b> suma	<b>-</b> resta
	<b>*</b> multiplicación	<b>/</b> división
<b>%</b>	resto de la división entera	
- **Relacionales**

<b>&gt;</b> mayor	<b>&lt;</b> menor
<b>&gt;=</b> mayor o igual	<b>&lt;=</b> menor o igual
<b>!=</b> distinto	<b>==</b> igual
- **Lógicos**

<b>&amp;&amp;</b> AND	<b>  </b> OR	<b>!</b> NOT
-----------------------	--------------	--------------

# EXPRESIONES

- Definición:  
Conjunto de variables, constantes, números y operadores ordenados de acuerdo a las reglas sintácticas establecidas en el lenguaje de programación.

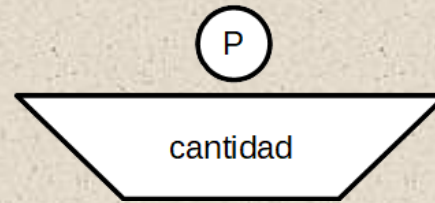
Tienen como objetivo la construcción de instrucciones para la resolución del problema (o de parte del problema) planteado.

# **OTROS PROGRAMAS EJEMPLO**

- Hacer un programa para calcular e informar el importe de una venta, a partir del ingreso de una cantidad de unidades de alfajores. El comercio vende cada caja de 12 alfajores a \$1000, y cada alfajor suelto a \$ 100.

- Hacer un programa para calcular e informar el importe de una venta, a partir del ingreso de una cantidad de unidades de alfajores. El comercio vende cada caja de 12 alfajores a \$1000, y cada alfajor suelto a \$ 100.

```
cin>>cantidad;
```





- Hacer un programa para calcular e informar el importe de una venta, a partir del ingreso de una cantidad de unidades de alfajores. El comercio vende cada caja de 12 alfajores a \$1000, y cada alfajor suelto a \$ 100.

```
cin>>cantidad;
```

```
cout<<"El importe es: ";  
cout<<importe;
```

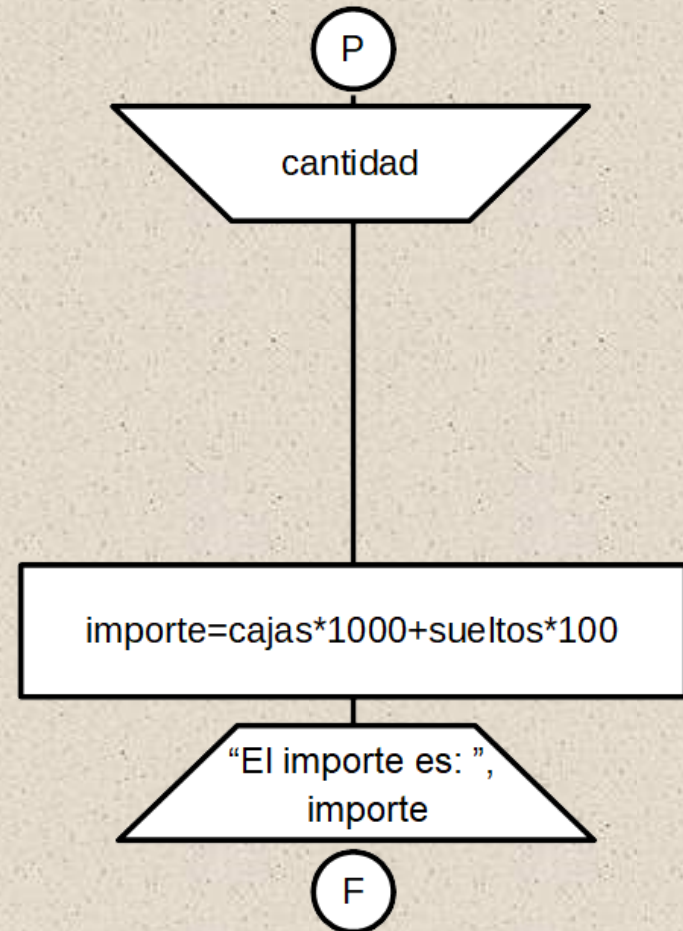


- Hacer un programa para calcular e informar el importe de una venta, a partir del ingreso de una cantidad de unidades de alfajores. El comercio vende cada caja de 12 alfajores a \$1000, y cada alfajor suelto a \$ 100. .

```
cin>>cantidad;
```

```
importe=cajas*1000+  
suelos *100;
```

```
cout<<"El importe es: ";  
cout<<importe;
```



- Hacer un programa para calcular e informar el importe de una venta, a partir del ingreso de una cantidad de unidades de alfajores. El comercio vende cada caja de 12 alfajores a \$1000, y cada alfajor suelto a \$ 100.

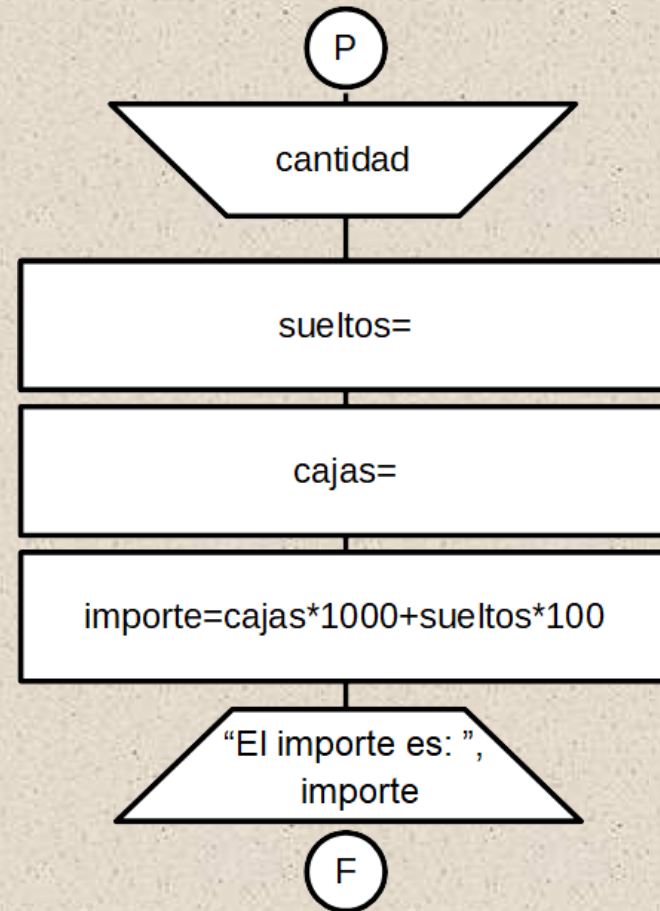
```
cin>>cantidad;
```

```
suelos=;
```

```
cajas=;
```

```
importe=cajas*1000+  
suelos *100;
```

```
cout<<"El importe es: ";  
cout<<importe;
```



- Hacer un programa para calcular e informar el importe de una venta, a partir del ingreso de una cantidad de unidades de alfajores. El comercio vende cada caja de 12 alfajores a \$1000, y cada alfajor suelto a \$ 100.

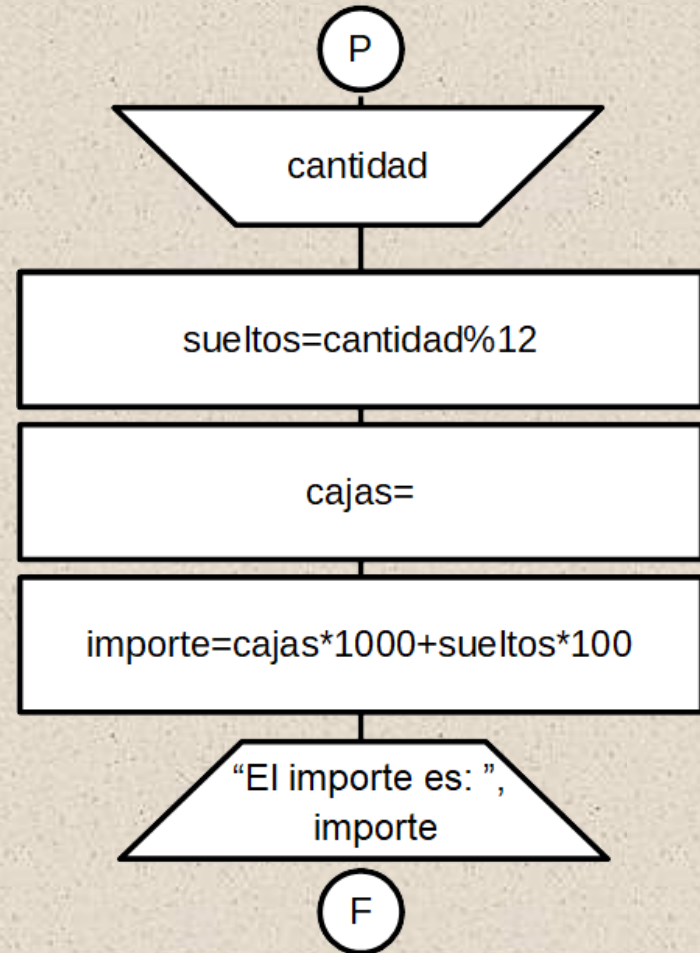
```
cin>>cantidad;
```

```
suektos=cantidad%12;
```

```
cajas=;
```

```
importe=cajas*1000+  
suektos *100;
```

```
cout<<"El importe es: ";  
cout<<importe;
```



- Hacer un programa para calcular e informar el importe de una venta, a partir del ingreso de una cantidad de unidades de alfajores. El comercio vende cada caja de 12 alfajores a \$1000, y cada alfajor suelto a \$ 100

```
cin>>cantidad;
```

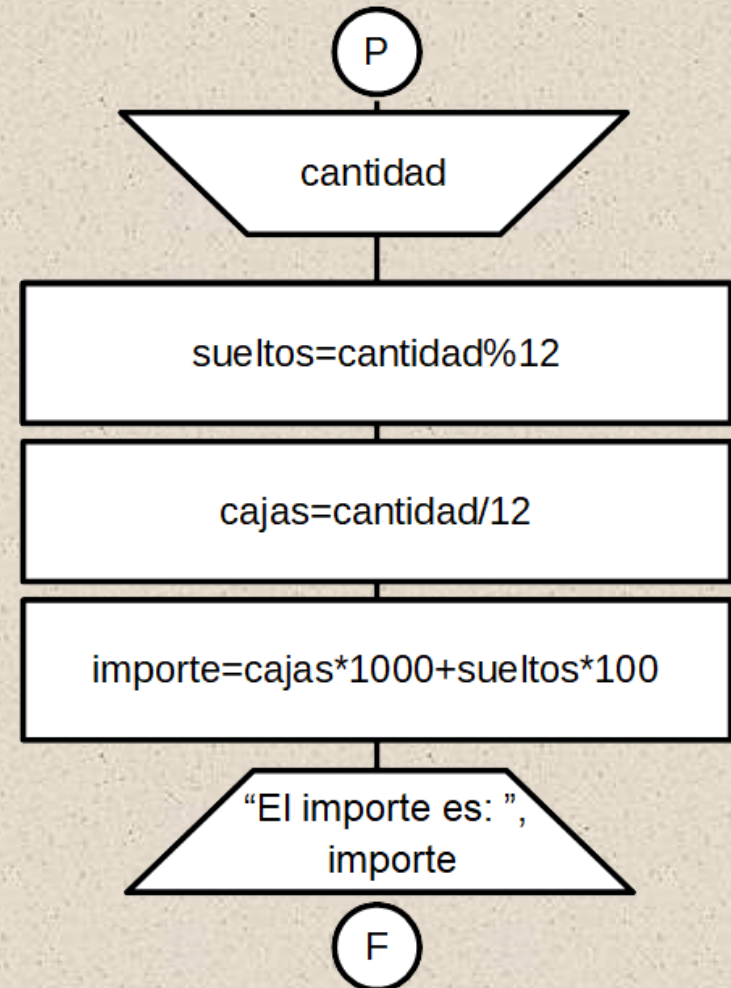
```
suelos=cantida%12;
```

```
cajas=cantidad/12;
```

```
importe=cajas*1000+suelos *100;
```

```
cout<<"El importe es: ";
```

```
cout<<importe;
```

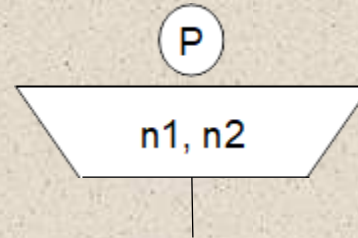




- Dados 2 números calcular e informar el mayor de ellos

- Dados 2 números calcular e informar el mayor de ellos

```
cin>>n1;  
cin>>n2;
```

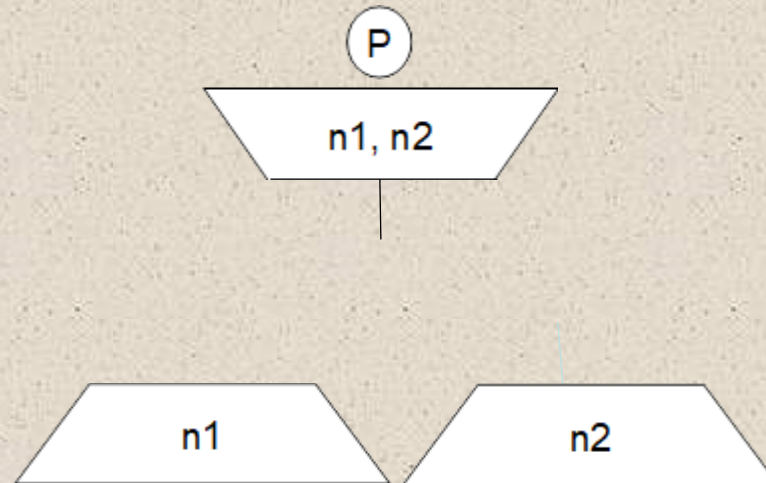


- Dados 2 números calcular e informar el mayor de ellos

```
cin>>n1;  
cin>>n2;
```

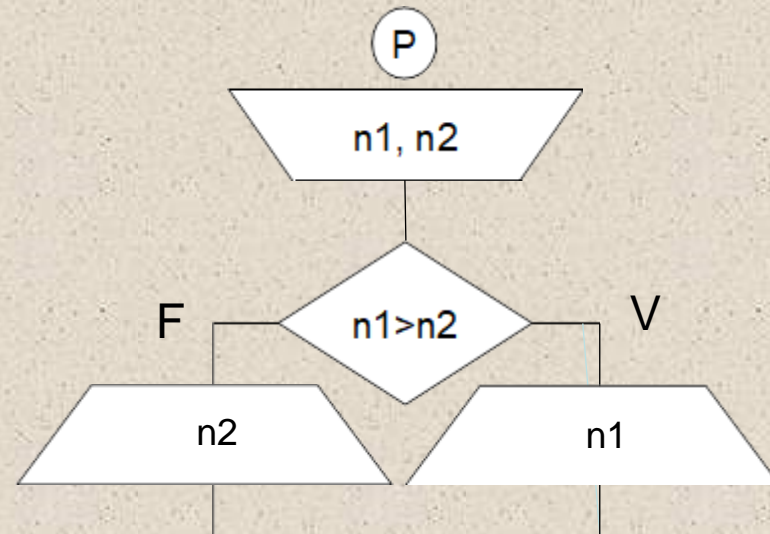
```
cout<<n1;
```

```
cout<<n2;
```

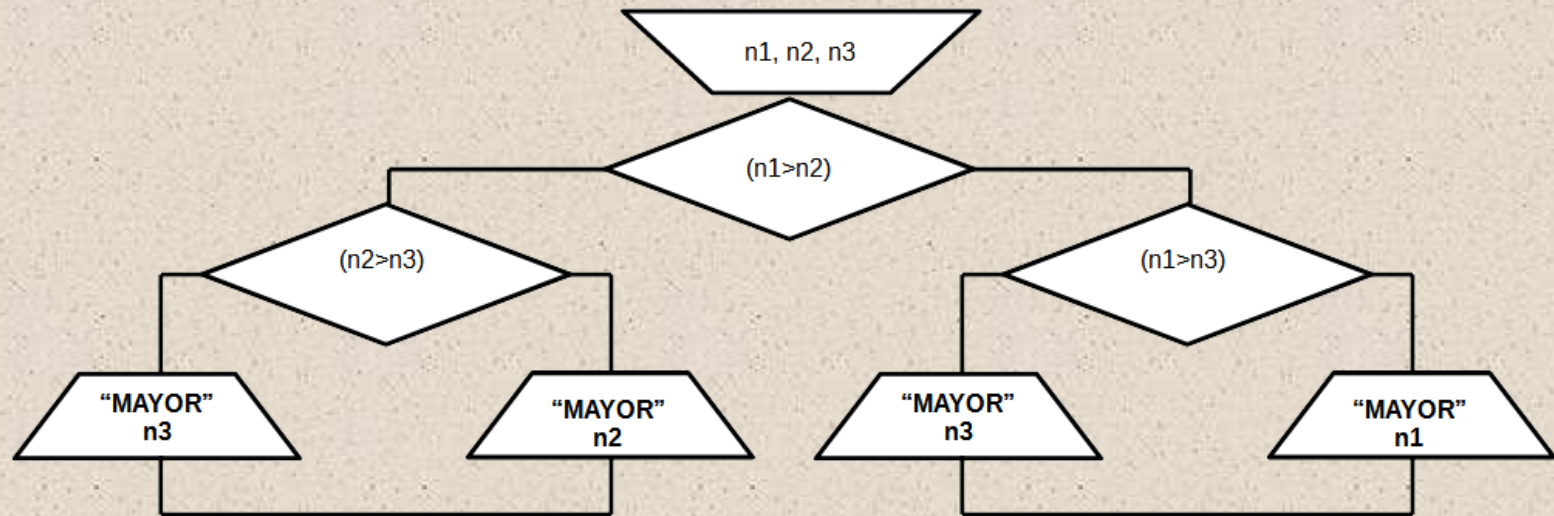


- Dados 2 números calcular e informar el mayor de ellos

```
int main(){  
    int n1,n2;  
    cin>>n1;  
    cin>>n2;  
    if(n1>n2){  
        cout<<n1;  
    }  
    else{  
        cout<<n2;  
    }  
    return 0;  
}
```



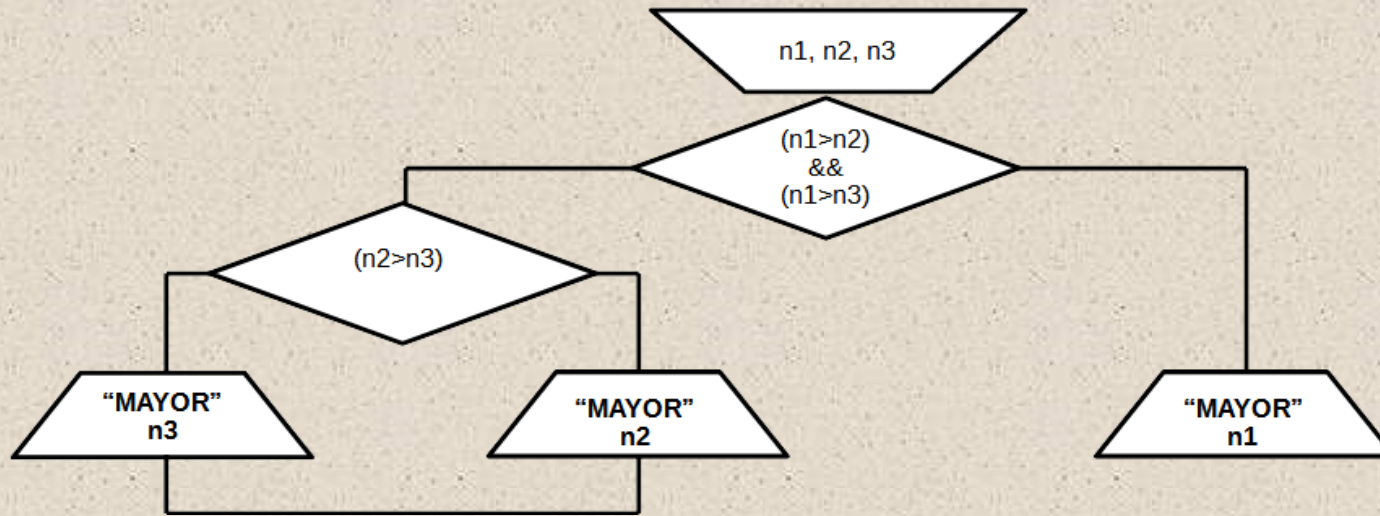
- Dados 3 números calcular e informar el mayor de ellos



```
int main(){
    int n1,n2, n3;
    cin>>n1;
    cin>>n2;
    cin>>n3;
    if(n1>n2){
        if(n1>n3) {cout<<"MAYOR: "<<n1;}
        else { cout<<"MAYOR: "<<n3;}
    }
    else {
        if(n2>n3){ cout<<"MAYOR: "<<n2;}
        else {cout<<"MAYOR: "<<n3;}
    }
    return 0;
}
```

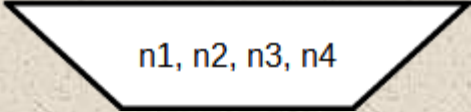


- Dados 3 números calcular e informar el mayor de ellos



```
int main(){
    int n1,n2, n3;
    cin>>n1;
    cin>>n2;
    cin>>n3;
    if(n1>n2 && n1>n3){
        cout<<"MAYOR: "<<n1;
    }
    else {
        if(n2>n3){ cout<<"MAYOR: "<<n2;}
        else {cout<<"MAYOR: "<<n3;}
    }
    return 0;
}
```

- Dados 4 números calcular e informar el mayor de ellos



n1, n2, n3, n4

- Dados 4 números calcular e informar el mayor de ellos

The diagram illustrates a process for finding the maximum of four numbers. At the top, a downward-pointing trapezoid represents the input, containing the text "n1, n2, n3, n4". Below this, four upward-pointing trapezoids represent the output, each containing the text "“MAYOR”" followed by a subscripted number. From left to right, these are labeled "n4", "n3", "n2", and "n1", indicating that the maximum value is identified and reported for each of the four input numbers.

n1, n2, n3, n4

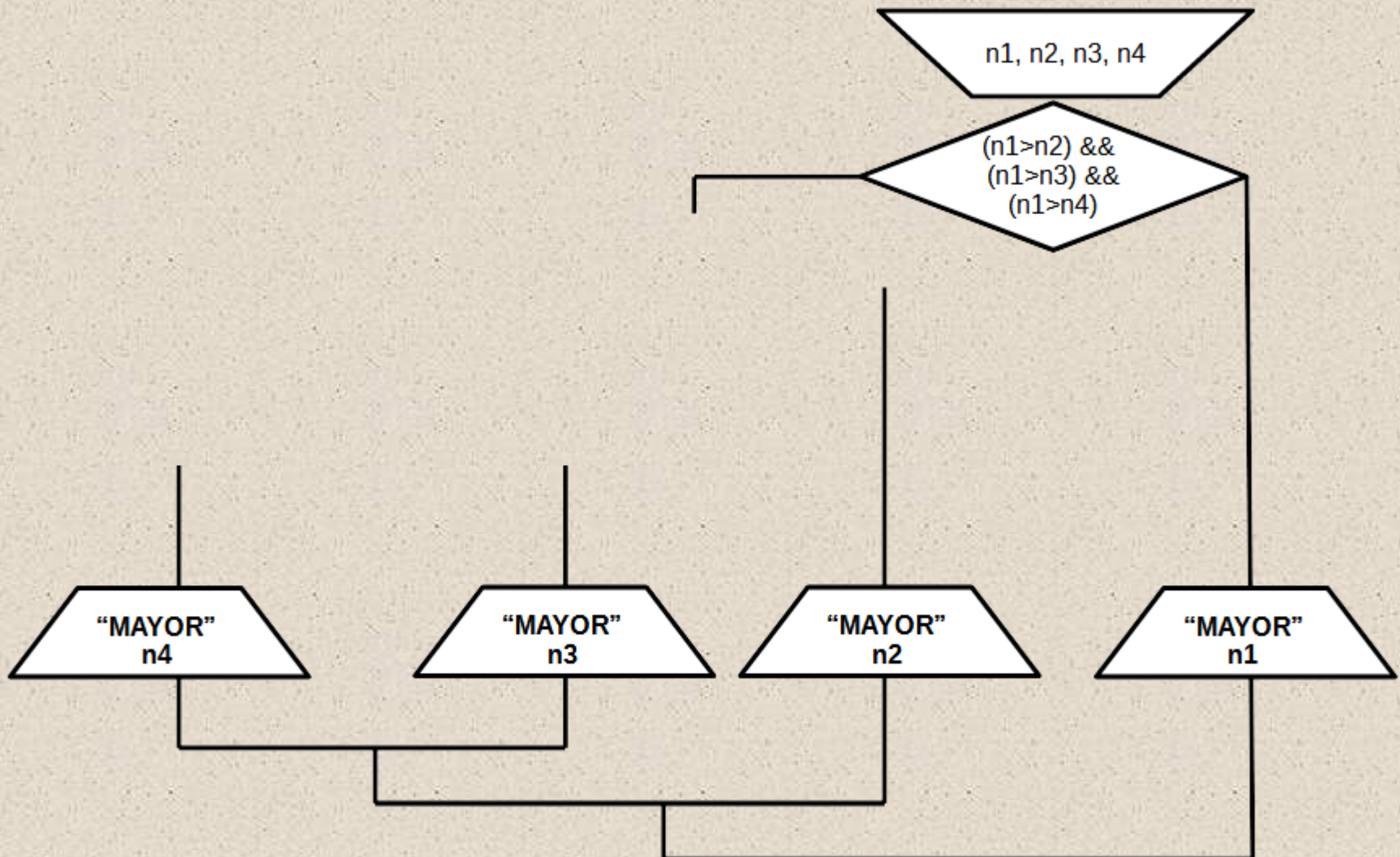
“MAYOR”  
n4

“MAYOR”  
n3

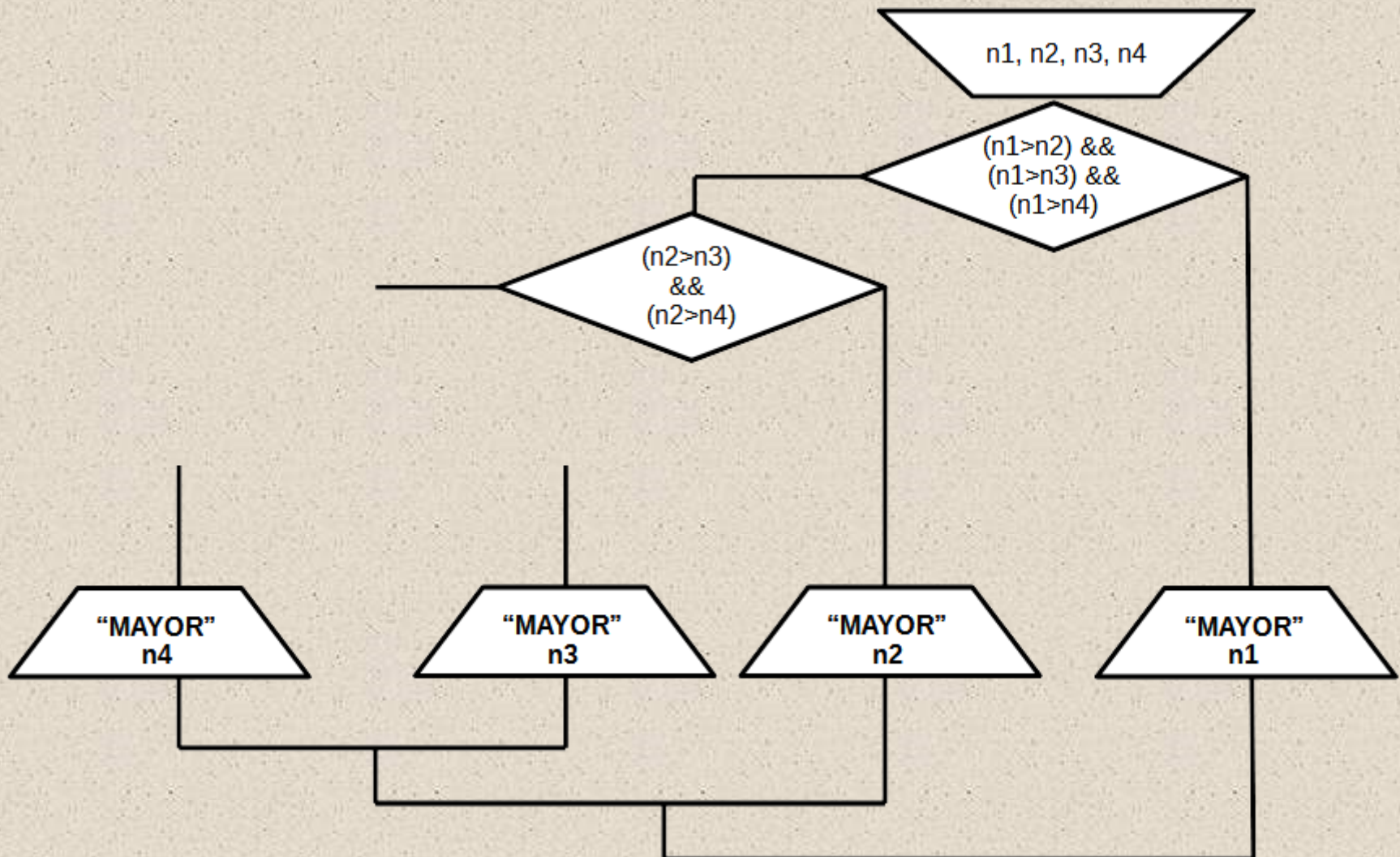
“MAYOR”  
n2

“MAYOR”  
n1

- Dados 4 números calcular e informar el mayor de ellos

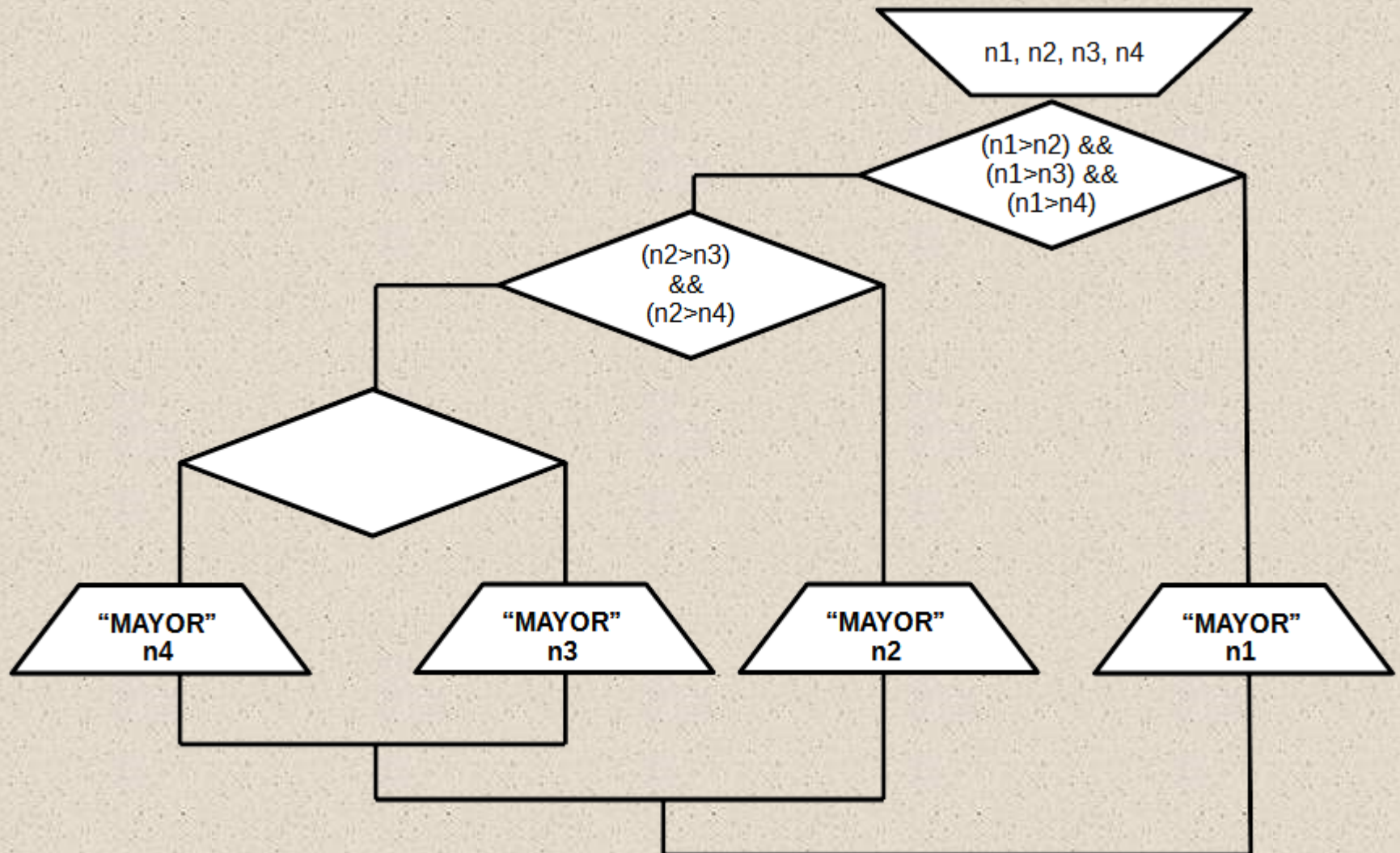


- Dados 4 números calcular e informar el mayor de ellos

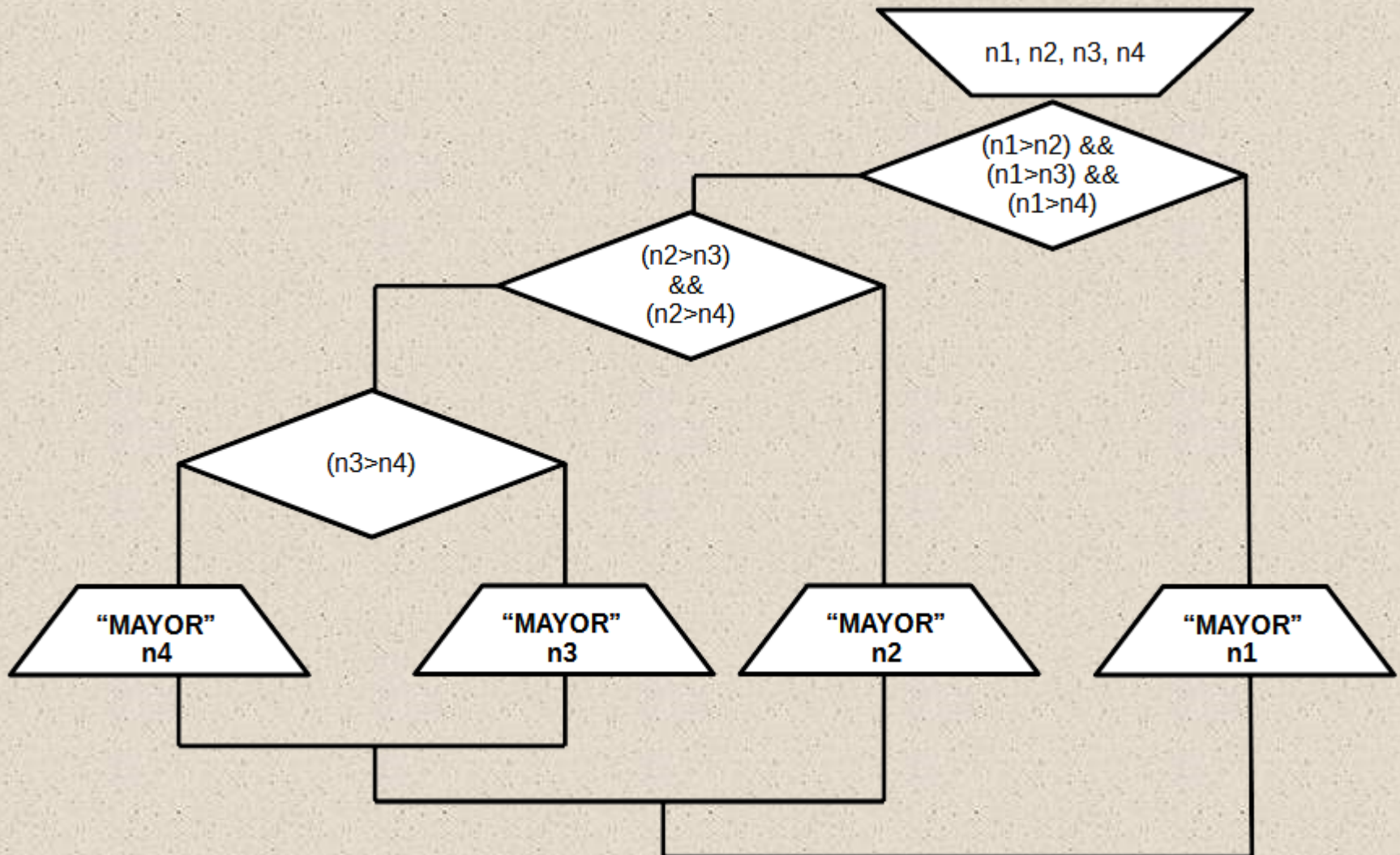




- Dados 4 números calcular e informar el mayor de ellos

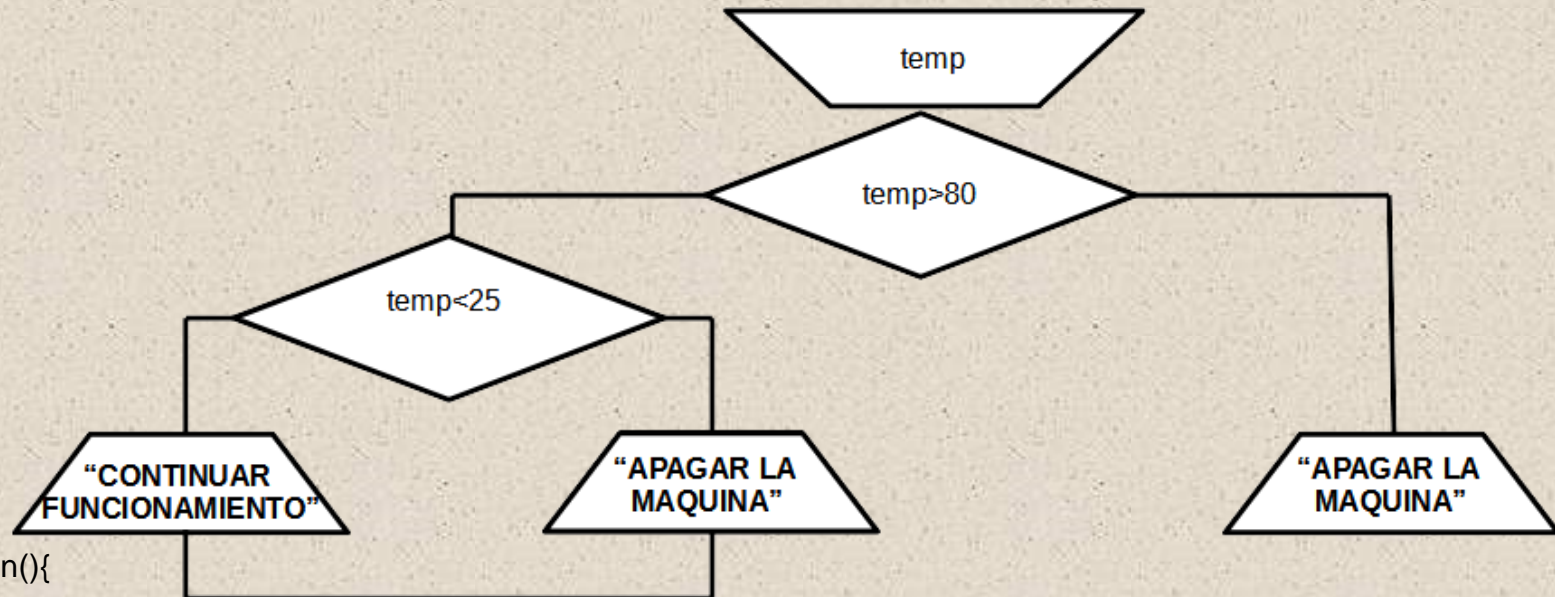


- Dados 4 números calcular e informar el mayor de ellos



- Una máquina acepta una temperatura de trabajo entre 25 y 80 grados, incluyendo ambos extremos. Si la temperatura baja de 25, o sube a más de 80 debe apagarse.

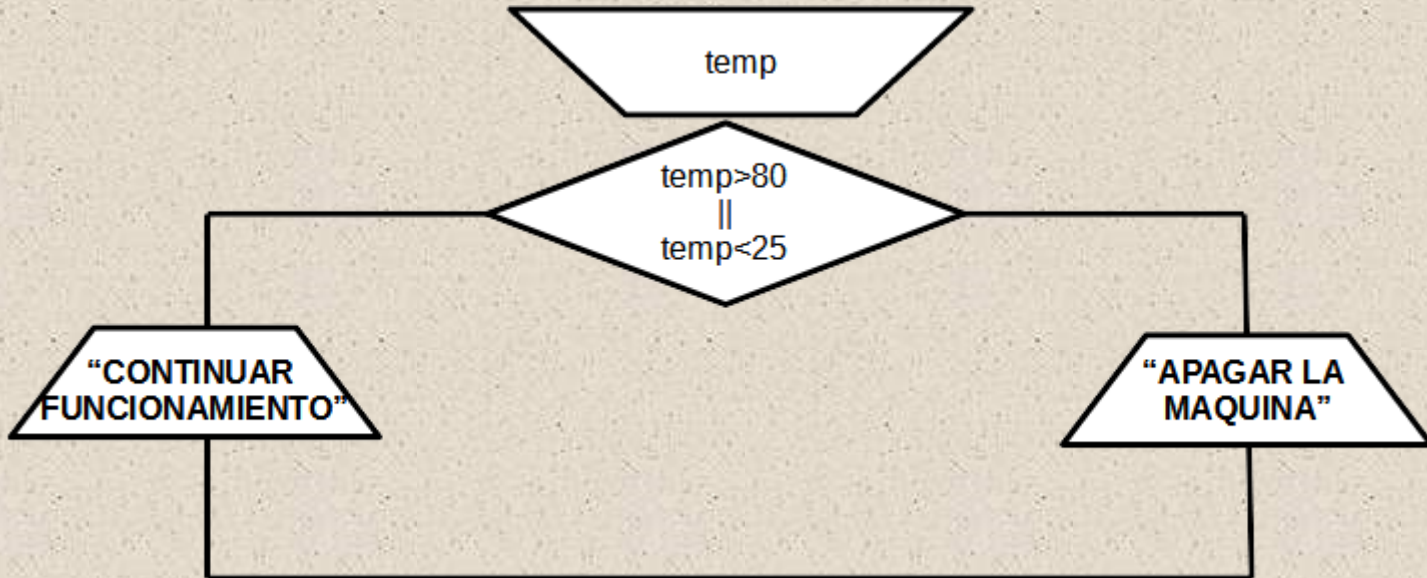
Hacer un programa para que al ingresar un valor de temperatura se informe si debe apagarse, o continuar funcionando.



```
int main(){
    int temp;
    cin>>temp;
    if(temp>80){
        cout<<"APAGAR LA MAQUINA";
    }
    else {
        if(temp<25){ cout<<"APAGAR LA MAQUINA";}
        else {cout<<"CONTINUAR FUNCIONAMIENTO";}
    }
    return 0;
}
```

- Una máquina acepta una temperatura de trabajo entre 25 y 80 grados, incluyendo ambos extremos. Si la temperatura baja de 25, o sube a más de 80 debe apagarse.

Hacer un programa para que al ingresar un valor de temperatura se informe si debe apagarse, o continuar funcionando.



```
int main(){
    int temp;
    cin>>temp;
    if(temp>80 || temp<25){
        cout<<"APAGAR LA MAQUINA";
    }
    else {
        cout<<"CONTINUAR FUNCIONAMIENTO";
    }
    return 0;
}
```

# MATERIAL DE TRABAJO

- En el Aula Virtual de la materia:
  - Apuntes teóricos.
  - Guías de ejercicios
- - Videos