

### -Ejercicio 1) Parciales

Dadas las notas de 3 parciales, calcular promedio y decir si promocionó o rinde final.

## Analisis

**entradas:** nota1, nota2, nota3

**salidas:** promedio, condicion

```
procesos: promedio <- (nota1+ nota2+ nota3)/3
           condicion <- si nota <= 7 entonces promociono
                       sino rinde final
```

## Ambiente

variable	tipo	descripcion
nota1	real	nota del primer parcial
nota2	real	nota del segundo parcial
nota3	real	nota del tercer parcial
promedio	real	promedio de los tres parciales
condicion	caracter	condición del alumno

## Estrategia

- ingresar las tres notas de los parciales
- calcular el promedio
- analizar la condicion del alumno
- mostrar resultado en pantalla

## Algoritmo

```

Proceso notaparcial
    definir nota1, nota2, nota3, promedio como real;

    Escribir "ingrese la primer nota del parcial";
    leer nota1;
    Escribir "ingrese la segunda nota del parcial";
    leer nota2;
    Escribir "ingrese la tercer nota del parcial";
    leer nota3;

    promedio <- (nota1+ nota2+ nota3)/3 ;
    Escribir "el promedio del alumno es: ", promedio;

    si promedio <= 7 Entonces
        escribir " la condicion el alumno es: rendir final";
    sino
        Escribir "la condicion el alumno es: aprobacion directa";
    FinSi
FinProceso

```

## Prueba de escritorio

nota1	nota2	nota3	promedio	condicion	salidas
					ingrese las 3 notas de los parciales
5	6	9			
5	6	9	6,66		
5	6	9	6,66	rinde final	el alumno rendira final

### Ejercicio 2) Par o impar

Se necesita un algoritmo que informe si un número ingresado es PAR o IMPAR mediante un mensaje.

## Analisis

**entradas:** num

**salidas:** es par o es impar

**procesos:** si  $\text{num} \bmod 2 == 0$  entonces "es par"  
sino "es impar"

## Ambiente

variable	tipo	descripcion
num	entero	numero que se ingresa

## Estrategia

```

ingresar un num
evaluar condicion si
    informar "es par" *
    informar "esimpar" *

```

## Algoritmo

```
Proceso parimpar
    definir num como entero;

    escribir "ingrese un numero";
```

```

    leer num;

    si num mod 2 == 0 Entonces
        escribir "el numero es par";
    sino
        escribir "el numero es impar";
    FinSi

FinProceso

```

### Prueba de escritorio

num	salidas
	ingrese un numero
8743	el numero es impar

### Ejercicio 3) Positivo, negativo o cero

Se desea saber si el número ingresado es positivo, negativo o cero.

#### Analisis

**entradas:** num

**salidas:** indicar si num es positivo, negativo o cero

**procesos:** si num == 0 entonces num es 0  
 sino  
 si num > 0 entonces num es positivo  
 sino num es negativo

#### Ambiente

variable	tipo	descripcion
num	entero	numero que se ingresa

#### Estrategia

```

ingresar un numero
verificar cero *
    es cero *
    verificar positivo o negativo*
        es positivo*
        es negativo*

informar en pantalla

```

## Algoritmo

Proceso posinegacero

definir num como entero;

escribir "ingrese un numero";

leer num;

si num == 0 Entonces

escribir "el numero es cero";

sino

si num > 0 Entonces

escribir "el numero es positivo";

sino

Escribir "el numero es negativo";

FinSi

FinSi

FinProceso

## Prueba de escritorio

num	salidas
	ingrese un numero
0	es cero

## Ejercicio 4) Triangulo

Se requiere de un algoritmo que permita determinar si 3 segmentos de recta pueden formar un triángulo.

*Nota: En cualquier triangulo el mayor de los lados es menor que la suma de los restantes, o en general, la suma de 2 lados debe ser mayor que el lado restante.*

### Analisis

**entradas:** num1,num2,num3, mayor

**salidas:** informar si se puede formar un triangulo o no

**procesos:** mayor = (num1,num2,num3)

si mayor < (num1+num2+num3-mayor) entonces informar "se puede formar un triangulo"

sino informar "no se puede formar un triangulo"

### Ambiente

nombre	tipo	descripcion
num1/2/3	entero	numeros ingresados

mayor	entero	numero mayor del conjunto
-------	--------	---------------------------

### Estrategia

```

ingresar primer numero
actualizar mayor
ingresar el segundo numero
verificar mayor*
    actualizar mayor*
/*
ingresar el tercer numero
verificar mayor*
    actualizar mayor*
/*

verificar condicion del mayor*
    informar "hay triangulo"*
    informar "no hay triangulo"*

```

### Algoritmo

```

Proceso posinegacero
    definir num1, num2, num3, mayor como entero;

    escribir "ingrese el 1er numero";
    leer num1;
    mayor <- num1;

    escribir "ingrese el 2do numero";
    leer num2;

    si mayor < num2 Entonces
        mayor <- num2;
    FinSi

    escribir "ingrese el 3er numero";
    leer num3;

    si mayor < num3 Entonces
        mayor <- num3;
    FinSi

    si mayor < (num1+num2+num3-mayor) entonces
        escribir "se puede formar un triangulo";
    sino
        escribir "no se puede formar un triangulo";
    FinSi

FinProceso

```

### Prueba de escritorio

num1	num2	num3	mayor	salidas
				ingrese un numero
2			2	ingrese un numero
2	3		3	ingrese un numero
2	3	6	6	no se puede formar un triangulo

### Ejercicio 5) Números de libreta

Se ingresa el nombre y Nro de libreta de 3 alumnos. Muestre la lista ordenada por Nro de libreta.

#### Analisis

**entradas:** L1,N1,L2, N2, L3, N3

**salidas:** lista ordenada por numero de libreta y nombre de menor a mayor

**procesos:** si  $L1 < L2$  y  $L1 < L3$  entonces

    si  $L2 < L3$  entonces

        informar L1, N1

        L2, N2

        L3, N3

    sino

        informar L1, N1

        L3, N3

        L2, N2

    si  $L2 < L1$  y  $L2 < L3$  entonces

    si  $L1 < L3$  entonces

        informar L2, N2

        L1, N1

        L3, N3

    sino

        informar L2, N2

        L3, N3

        L1, N1

    si  $L3 < L1$  y  $L3 < L2$  entonces

    si  $L1 < L2$  entonces

        informar L3, N3

        L1, N1

        L2, N2

    sino

        informar L3, N3

L2, N2  
L1, N1

## Ambiente

nombre	tipo	descripcion
N1/2/3	caracter	nombre de los alumnos
L1/2/3	entero	numero de libreta

## Estrategia

ingresar las libretas y los numeros  
verificar el 1ero como menor<sup>o</sup>

verificar intermedio<sup>o</sup>  
ordenar y mostrar<sup>o</sup>  
ordenar y mostrar<sup>o</sup>

verificar el 2do como menor<sup>o</sup>

verificar intermedio<sup>o</sup>  
ordenar y mostrar<sup>o</sup>  
ordenar y mostrar<sup>o</sup>

verificar el 3er como menor<sup>o</sup>  
/\*  
verificar intermedio<sup>o</sup>  
ordenar y mostrar<sup>o</sup>  
ordenar y mostrar<sup>o</sup>

## Algoritmo

Proceso libretas

Definir L1,L2,L3 Como Entero;  
Definir N1,N2,N3 Como Caracter;  
Escribir 'ingresar el primer nombre y libreta';  
Leer N1;  
Leer L1;  
Escribir 'ingresar el segundo nombre y libreta';  
Leer N2;  
Leer L2;  
Escribir 'ingresar el tercer nombre y libreta';  
Leer N3;  
Leer L3;  
Si L1<L2 Y L1<L3 Entonces  
Si L2<L3 Entonces  
Escribir L1,N1;  
Escribir L2,N2;

```

        Escribir L3,N3;
    SiNo
        Escribir L1," ", N1;
        Escribir L3," ", N3;
        Escribir L2," ", N2;
    FinSi
SiNo
    Si L2<L1 Y L2<L3 Entonces
        Si L1<L3 Entonces
            Escribir L2," ", N2;
            Escribir L1," ", N1;
            Escribir L3," ", N3;
        SiNo
            Escribir L2," ", N2;
            Escribir L3," ", N3;
            Escribir L1," ", N1;
        FinSi
    SiNo
        Si L3<L1 Y L3<L2 Entonces
            Si L1<L2 Entonces
                Escribir L3," ", N3;
                Escribir L1," ", N1;
                Escribir L2," ", N2;
            SiNo
                Escribir L3," ", N3;
                Escribir L2," ", N2;
                Escribir L1," ", N1;
            FinSi
        FinSi
    FinSi
FinProceso

```

### Prueba de escritorio

N1	L1	N2	L2	N3	L3	salidas
						ingese el primer nombre y numero
pedrito	17999					ingese el segundo nombre y numero
pedrito	17999	juancito	17884			ingese el tercer nombre y numero
pedrito	17999	juancito	17884	manuel	17228	



pedrito	17999	juancito	17884	manuel	17288	mostrar en pantalla: 17288 manuel 17884 juancito 17999 pedrito
---------	-------	----------	-------	--------	-------	--

### Ejercicio 6) Rectángulo

Realice un algoritmo que, tomando como datos la base y la altura de un rectángulo, informe si este es horizontal o vertical. Sin dejar de considerar el caso particular del cuadrado. Finalmente calcule el área de la figura.

#### Analisis

**entradas:** base, altura

**salidas:** informar si es un rectangulo horizontal, vertical o un cuadrado  
area

**procesos:** si base == altura entonces informar "es un cuadrado"  
sino  
si base > altura entonces informar "es un rectangulo horizontal"  
sino  
informar "es un rectangulo vertical"  
  
area<- base\*altura

#### Ambiente

nombre	tipo	descripcion
base	entero	base de la figura
altura	entero	altura de la figura
area	entero	area de la figura

#### Estrategia

ingresar la base y altura  
verificar si es cuadrado<sup>o</sup>  
informar es cuadrado<sup>o</sup>  
verificar horizontal<sup>o</sup>  
informar es horizontal<sup>o</sup>  
informar es vertical<sup>o</sup>  
  
calcular area  
mostrar en pantalla

## Algoritmo

Proceso cuadrado

definir base, altura, area como entero;

escribir "ingrese en centimetros la base y altura";

leer base, altura;

si base == altura entonces

escribir "es un cuadrado";

sino

si base > altura entonces

escribir "es un rectangulo horizontal";

sino

escribir "es un rectangulo vertical";

fin si

FinSi

area <- base\*altura;

escribir "el area de la figura es: ", area, " cm";

FinProceso

## Prueba de escritorio

base	altura	area	salidas
			ingrese la base y altura
8	8	64	
8	8	64	es un cuadrado, tiene 64cm de area

## Ejercicio 7) Mayor valor

Realice un algoritmo que pida 5 valores al usuario y luego informe cual es el mayor de los ingresados.

Restricción: la aplicación solo puede tener 2 variables.

### Analisis

**entradas:** num

**salidas:** mayor

**procesos:** mayor <- (numero que pertenece a los del conjunto ingresado)

### Ambiente

nombre	tipo	descripcion
--------	------	-------------

num	entero	numero ingresado
mayor	entero	mayor numero de los ingresados

### Estrategia

ingresar el primer valor  
 actualizar mayor  
 ingreso el segundo valor  
 verifico mayor<sup>o</sup>  
     actualizo mayor<sup>o</sup>  
     /<sup>o</sup>  
 ingreso el tercer valor  
 verifico mayor<sup>o</sup>  
     actualizo mayor<sup>o</sup>  
     /<sup>o</sup>  
 ingreso el cuarto valor  
 verifico mayor<sup>o</sup>  
     actualizo mayor<sup>o</sup>  
     /<sup>o</sup>  
 ingreso el quinto valor  
 verifico mayor<sup>o</sup>  
     actualizo mayor<sup>o</sup>  
     /<sup>o</sup>  
 informar mayor

### Algoritmo

Proceso maior  
     definir num, mayor como entero;  
  
     escribir "ingrese el primer valor";  
     leer num;  
     mayor<- num;  
     escribir "ingrese el segundo valor";  
     leer num;  
  
     si num>mayor Entonces  
         mayor<- num;  
     FinSi  
  
     escribir "ingrese el tercer valor";  
     leer num;  
  
     si num>mayor Entonces  
         mayor<- num;  
     FinSi  
  
     escribir "ingrese el cuarto valor";  
     leer num;  
  
     si num>mayor Entonces

```

        mayor<- num;
    FinSi

    escribir "ingrese el quinto valor";
    leer num;

    si num>mayor Entonces
        mayor<- num;
    FinSi

    Escribir "el mayor de los ingresados es el numero: ", mayor;
FinProceso

```

### Prueba de escritorio

num	mayor	descripcion
3	3	ingrese un valor
6	6	ingrese el siguiente valor
12	12	ingrese el siguiente valor
4	12	ingrese el siguiente valor
38	38	ingrese el siguiente valor
37	38	el numero mayor es 38

### Ejercicio 8) Orden que ocurrió el menor

Realice un algoritmo que pida 5 valores al usuario y luego informe cual es el número de orden en que se ingresó el menor de ellos.

Restricción: la aplicación solo puede tener 3 variables.

#### Analisis

**entradas:** num

**salidas:** menor, orden

**procesos:** menor <- (numero que pertenece al conjunto de los ingresados)  
orden <- (posicion en la ue se ingresa el menor de los numeros)

#### Ambiente

nombre	tipo	descripcion
num	entero	numero ingresado
menor	entero	menor numero de los ingresados

orden	entero	orden en el que ingresa el menor de ellos
-------	--------	---

### Estrategia

ingresar el primer valor  
 actualizar menor  
 ingreso el segundo valor  
 verifico menor<sup>o</sup>  
     actualizo menor y orden<sup>o</sup>  
     /<sup>o</sup>  
 ingreso el tercer valor  
 verifico menor<sup>o</sup>  
     actualizo menor y orden<sup>o</sup>  
     /<sup>o</sup>  
 ingreso el cuarto valor  
 verifico menor<sup>o</sup>  
     actualizo menor y orden<sup>o</sup>  
     /<sup>o</sup>  
 ingreso el quinto valor  
 verifico menor<sup>o</sup>  
     actualizo menor y orden<sup>o</sup>  
     /<sup>o</sup>  
  
 informar menor

### Algoritmo

Proceso menorcito  
     definir num, menor, orden como entero;  
  
     escribir "ingrese el primer valor";  
     leer num;  
     menor<- num;  
     orden<-1;  
     escribir "ingrese el segundo valor";  
     leer num;  
  
     si num<menor Entonces  
         menor<- num;  
         orden <- 2;  
     FinSi  
  
     escribir "ingrese el tercer valor";  
     leer num;  
  
     si num<menor Entonces  
         menor<- num;  
         orden <- 3;  
     FinSi  
  
     escribir "ingrese el cuarto valor";  
     leer num;

```

    si num<menor Entonces
        menor<- num;
        orden <- 4;
    FinSi

    escribir "ingrese el quinto valor";
    leer num;

    si num<menor Entonces
        menor<- num;
        orden <- 5;
    FinSi
    Escribir "el menor de los ingresados es el numero: ", menor, " en el orden nro: ",
orden;
FinProceso

```

### Prueba de escritorio

num	menor	orden	descripcion
3	3	1	ingrese un valor
6	3	1	ingrese el siguiente valor
12	3	1	ingrese el siguiente valor
4	3	1	ingrese el siguiente valor
38	3	1	ingrese el siguiente valor
37	3	1	el numero menor es 3 en el orden 1

### Ejercicio 9) Tenis

Ingresado el nombre de los jugadores y el resultado de cada set (3) de un partido de tenis, informe en pantalla cual es el ganador.

*Ejemplo: Nadal, Del Potro: 7,5,4,6,6,2*

*Ganador Nadal*

### Analisis

**entradas:** N1, N2, PJ1, PJ2

**salidas:** ganador

**procesos:** ganador<- verifico contador de set ganados; si contj1 > contj2 entonces  
informar "el ganador es el jugador 1"  
sino  
informar "el ganador es el jugador 2"

## Ambiente

nombre	tipo	descripcion
<b>N1/2</b>	<b>caracter</b>	<b>nombre de los jugadores</b>
<b>PJ1/2</b>	<b>entero</b>	<b>puntos hechos en el set</b>
<b>contj1/2</b>	<b>entero</b>	<b>contador de set ganados</b>

## Estrategia

ingresar nombre de jugadores  
ingresar puntos de ambos jugadores del primer set  
verificar quien tiene mas puntos<sup>o</sup>  
    contador +1 para jugador 1<sup>o</sup>  
    contador +1 para jugador 2<sup>o</sup>  
ingresar puntos de ambos jugadores del segundo set  
verificar quien tiene mas puntos<sup>o</sup>  
    contador +1 para jugador 1<sup>o</sup>  
    contador +1 para jugador 2<sup>o</sup>  
ingresar puntos de ambos jugadores del tercer set  
verificar quien tiene mas puntos<sup>o</sup>  
    contador +1 para jugador 1<sup>o</sup>  
    contador +1 para jugador 2<sup>o</sup>  
  
verificar contadores<sup>o</sup>  
    el ganador es el jugador 1<sup>o</sup>  
    el ganador es el jugador 2<sup>o</sup>

## Algoritmo

Proceso tenis  
    definir PJ1,PJ2,CONTJ1, CONTJ2 como entero;  
    definir N1,N2 Como Caracter;  
  
    escribir "ingresar el nombre de los jugadores";  
    leer N1, N2;  
  
    Escribir "ingrese los puntos del primer set ";  
    leer PJ1, PJ2;  
  
    si PJ1>PJ2 Entonces  
        CONTJ1 <- 1;  
    SiNo  
        CONTJ2<- 1;  
    FinSi

Escribir "ingrese los puntos del segundo set ";  
 leer PJ1, PJ2;

si PJ1>PJ2 Entonces  
     CONTJ1 <- +1;

SiNo  
     CONTJ2<- +1;

FinSi  
 Escribir "ingrese los puntos del tercer set ";  
 leer PJ1, PJ2;

si PJ1>PJ2 Entonces  
     CONTJ1 <- +1;

SiNo  
     CONTJ2<- +1;

FinSi

si CONTJ1>CONTJ2 Entonces  
     Escribir " el ganador es : ", N1;

sino  
     Escribir "el ganador es : ", N2;

FinSi

FinProceso

### Prueba de escritorio

N1	N2	PJ1	PJ2	CONTJ1	CONTJ2	SALIDAS
						ingresar los nombres de los jugadores
pedro	juan					ingresar puntos del primer set
pedro	juan	7	6	1	0	ingresar puntos del segundo set
pedro	juan	5	6	1	1	ingresar puntos del tercer set



<b>pedro</b>	<b>juan</b>	<b>4</b>	<b>2</b>	<b>2</b>	<b>1</b>	<b>el ganador es pedro</b>
--------------	-------------	----------	----------	----------	----------	----------------------------

### Ejercicio 10) Año bisiesto!

Implemente un algoritmo que permita determinar si un año es bisiesto o no.

Un año es bisiesto si es múltiplo de 4 (por ejemplo 1984). Los años múltiplos de 100 no son bisiestos, salvo si ellos son también múltiplos de 400 (2000 es bisiesto, pero; 1800 no lo es).

#### Analisis

**entradas:** anio

**salidas:** informar "es bisiesto" o "no es bisiesto"

**procesos:** si  $\text{anio} \bmod 4 == 0$  y ( $\text{anio} \bmod 100 <> 0$  o  $\text{anio} \bmod 400 == 0$ ) entonces informar "es bisiesto"  
sino "no es bisiesto"

#### Ambiente

nombre	tipo	descripcion
anio	entero	año que se analiza para saber si es o no bisiesto

#### Estrategia

ingresar el año

verificar si es bisiesto<sup>o</sup>

informar "es bisiesto"<sup>o</sup>

informar "no es bisiesto"<sup>o</sup>

#### Algoritmo

Proceso bisiesto

definir anio como entero;

Escribir "ingresar un año para determinar si es o no bisiesto";

leer anio;

si  $\text{anio} \bmod 4 == 0$  y ( $\text{anio} \bmod 100 <> 0$  o  $\text{anio} \bmod 400 == 0$ ) entonces

Escribir "el año ", anio, " es bisiesto";

SiNo

Escribir "el año ", anio, " no es bisiesto";

FinSi

FinProceso

#### Prueba de escritorio

anio	salidas
	ingresar un año
2020	es bisiesto

### Ejercicio 11) Día del mes

Realice un algoritmo que permita ingresar el número del mes y determine cuantos días tiene. Para el caso de Febrero, el algoritmo deberá indicar que no cuenta con la información necesaria para dar la respuesta.

#### Analisis

**entradas:** mes

**salidas:** informar cuantos dias tiene el mes

**procesos:** segun mes; 1,3,5,6,8,10,12: informar que el mes tiene 31 dias  
4,7,9,11: informar que el mes tiene 30 dias  
2: no se tiene la informacion necesaria

#### Ambiente

nombre	tipo	descripcion
mes	entero	mes que se analizara

#### Estrategia

ingresar el numero del algun mes

verificar a que mes corresponde<sup>o</sup>

informar "tiene 31 dias"<sup>o</sup>

informar "tiene 30 dias"<sup>o</sup>

informar "no se cuenta con la informacion necesaria"<sup>o</sup>

#### Algoritmo

Proceso bisiesto

definir anio como entero;

Escribir "ingresar el año para determinar si es bisiesto o no";

Leer anio;

Segun anio Hacer

1,3,5,6,8,10,12:

Escribir "el mes ingresado tiene 31 dias";

4,7,9,11:

Escribir "el mes ingresado tiene 30 dias";

2:

Escribir "no se cuenta con la informacion necesaria";

FinSegun

FinProceso

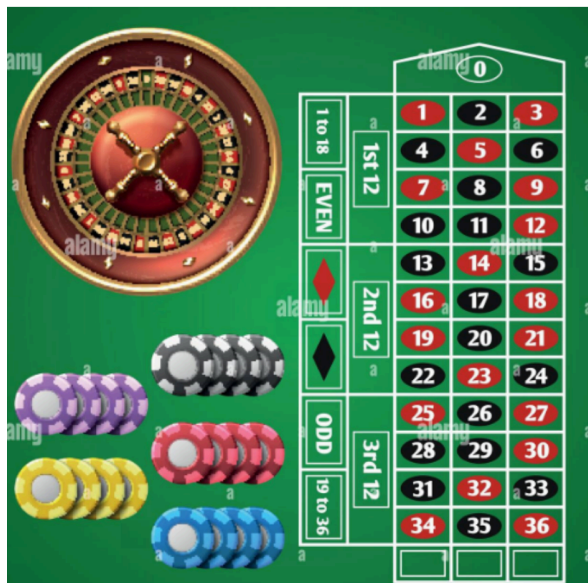
## Prueba de escritorio

mes	salidas
	ingrese el numero de un mes
8	tiene 31 dias

## Ejercicio 12) Ruleta

Se desea simular parte de un juego de ruleta donde el usuario ingresa un número entre 0 y 36 (el sistema debe verificarlo) y luego informar si es:

- 0 (banca gana)
- Mayor o Menor
- 1ra, 2da o 3ra Docena
- 1ra, 2da o 3ra Columna



## Analisis

**entradas:** num

**salidas:** a. 0 (banca gana)  
b. Mayor o Menor  
c. 1ra, 2da o 3ra Docena  
d. 1ra, 2da o 3ra Columna

**procesos:** gana banca <- si num == 0 entonces informar "gana banca"

mayor o menor <- si num <= 18 entonces informar "es menor" sino "es mayor"

1ra, 2da o 3ra Docena <- si num <= 12 entonces informar "pertenece a la primer docena"

```

sino
  si num <= 24 entonces informar "pertenece a la segunda docena"
sino
  informar "pertenece a la tercer docena"

1ra, 2da o 3ra columna<- si num mod 3 == 1 entonces informar "pertenece a
la primer columna"
sino
si num mod 3 == 2 entonces informar "el numero pertenece a la segunda columna"
sino informar "pertenece a la tercer columna"

```

## Ambiente

nombre	tipo	descripcion
num	entero	numero jugado

## Estrategia

```

1 ingresa el numero jugado
2 verificar numero valido°
  A      informar "numero no valido"
  B3     verificar cero°
    A      informar "gana banca"°
    B4     verificar mayor°
      A      informar "es mayor"°
      B      informar "es menor"°
      5      verificar primer docena °
        A      informar "pertenece a la primer docena"°
        B6     verificar segunda docena°
          A      informar "prtenece a la segDoc"°
          B      informar "prtenece a la terDoc"°

7 verificar primer columna°
  A      informar "pertenece a la primer columna"°
  B8     verificar segunda columna°
    A      informar "pertenece a la segunda columna"°
    B      informar "pertenece a la tercer columna"°

```

## Algoritmo

```

Proceso RULETA
  definir num como entero;
  escribir "ingrese un numero";
  leer num;
  si num >= 0 y num <= 36 entonces
    si num == 0 Entonces
      escribir "gana banca";
    SiNo
      si num <= 18 Entonces
        escribir "es menor";
      SiNo

```

```

        escribir "es mayor";

    FinSi
    si num <= 12 Entonces
        escribir "pertenece a la primer docena";
    sino
        si num <= 24 Entonces
            Escribir "pertenece a la segunda docena";
        sino
            Escribir "pertenece a la tercer docena";

        FinSi
    FinSi
    si num mod 3 == 1 Entonces
        escribir "pertenece a la primer columna ";
    SiNo
        si num mod 3 == 2 Entonces
            Escribir "pertenece a la segunda columna ";
        sino
            Escribir "pertenece a la tercer columna";
        FinSi
    FinSi
    FinSi
    SiNo
        Escribir "numero invalido";
    FinSi
FinProceso

```

### Prueba de escritorio

num	salidas
	<b>ingrese un numero</b>
<b>4</b>	<b>es menor</b>
<b>4</b>	<b>pertenece a la primer columna</b>
<b>4</b>	<b>pertenece a la primer docena</b>

### Ejercicio 13) Azar

Modifique el algoritmo anterior utilizando la función `Azar()` para generar un número aleatorio. ¿Qué modificaciones debe realizar?

### Algoritmo

Proceso RULETA

```

definir num como entero;
num <- azar(36);
escribir "el numero ingresado es ", num ;
si num >= 0 y num <= 36 entonces
    si num == 0 Entonces
        escribir "gana banca";
    SiNo
        si num <= 18 Entonces
            escribir "es menor";
        SiNo
            escribir "es mayor";

        FinSi
    si num <= 12 Entonces
        escribir "pertenece a la primer docena";
    sino
        si num <= 24 Entonces
            Escribir "pertenece a la segunda docena";
        sino
            Escribir "pertenece a la tercer docena";

        FinSi
    FinSi
    si num mod 3 == 1 Entonces
        escribir "pertenece a la primer columna ";
    SiNo
        si num mod 3 == 2 Entonces
            Escribir "pertenece a la segunda columna ";
        sino
            Escribir "pertenece a la tercer columna";
        FinSi
    FinSi
    FinSi
    SiNo
        Escribir "numero invalido";
    FinSi
FinProceso

```

**se utiliza la funcion azar() para que arroje un numero aleatorio de entre 0 y 36**

```

num <- azar(36);
    escribir "el numero ingresado es ", num ;

```