

**ALGORITMOS - PSEUDOCÓDIGO**

11.- Imaginemos que queremos desarrollar un algoritmo tal que al escribir un día de la semana aparezca por pantalla la actividad extraescolar a la que debe acudir nuestro hijo. Por ejemplo, si ejecutamos el programa y escribimos martes que aparezca "natación".

Las actividades que se realizan por días son:

lunes: psicomotricidad	jueves: natación
martes: natación	viernes: descanso
miércoles: música	

Los días sábado y domingo no se realizan actividades con lo que si el usuario escribe por error alguno de estos días de la semana se debe mostrar el mensaje: "Día sin actividades". Si por equivocación, además se escribe un día inexistente se debe mostrar en pantalla "Día erróneo".

```
1  Proceso ejercicio11
2      Escribir "Introduce día de la semana"
3      Leer dia
4      Segun dia Hacer
5          "lunes":
6              Escribir "psicomotricidad"
7          "martes", "jueves":
8              Escribir "natación"
9          "miércoles":
10             Escribir "música"
11         "viernes":
12             Escribir "descanso"
13         "sábado", "domingo":
14             Escribir "Día sin actividad"
15         De Otro Modo:
16             Escribir "Día erróneo"
17     Fin Segun
18 FinProceso
```

Tener en cuenta las mayúsculas, minúsculas y tildes. No es lo mismo lunes que Lunes, ni miércoles que miercoles. Esto es así porque el carácter 'l' es distinto del carácter 'L' en el código ASCII que es el que se usa para representarlos.

```
*** Ejecución Iniciada. ***
Introduce día de la semana
> miércoles
música
*** Ejecución Finalizada. ***
```

12.- Para realizar la **traza** se ha de hacer un seguimiento de los valores que toman las variables simulando manualmente la ejecución del algoritmo, es decir, siguiendo el flujo de ejecución del mismo.

Haz la traza del siguiente programa e indica qué valores toman las variables y qué escribe, suponiendo que a "a" se le da el valor 5:

```

1  Leer a
2  c <- 0
3  Mientras c<10 Hacer
4      c <- c+2
5      a <- a+3
6      Escribir a
7  Fin Mientras

```

Nº INSTRUCCIÓN	INSTRUCCIÓN	a	c	c<10	ESCRIBE
1	Leer a	5	-	-	-
2	c <- 0	5	0	-	-
3	Mientras c<10 Hacer	5	0	V	-
4	c <- c+2	5	2	-	-
5	a <- a+3	8	2	-	-
6	Escribir a	8	2	-	8
7	Fin Mientras	8	2	-	-
3	Mientras c<10 Hacer	8	2	V	-
4	c <- c+2	8	4	-	-
5	a <- a+3	11	4	-	-
6	Escribir a	11	4	-	11
7	Fin Mientras	11	4	-	-
3	Mientras c<10 Hacer	11	4	V	-
4	c <- c+2	11	6	-	-
5	a <- a+3	14	6	-	-
6	Escribir a	14	6	-	14
7	Fin Mientras	14	6	-	-
3	Mientras c<10 Hacer	14	6	V	-
4	c <- c+2	14	8	-	-
5	a <- a+3	17	8	-	-
6	Escribir a	17	8	-	17
7	Fin Mientras	17	8	-	-
3	Mientras c<10 Hacer	17	8	V	-
4	c <- c+2	17	10	-	-
5	a <- a+3	20	10	-	-
6	Escribir a	20	10	-	20
7	Fin Mientras	20	10	-	-
3	Mientras c<10 Hacer	20	10	F	-

13.- Realiza un algoritmo que nos indique si un año determinado es bisiesto, debemos pedir al usuario el año. Un año es bisiesto si es múltiplo de 4, a excepción de los que siendo múltiplos de 4 también son múltiplos de 100 pero no de 400.

```
1  Proceso ejercicio13
2      Definir anio Como Entero
3      Escribir "Introduce un año"
4      Leer anio
5      Si anio%4=0 Entonces
6          Si anio%100=0 y anio%400<>0 Entonces
7              Escribir anio, " no es bisiesto"
8          Sino
9              Escribir anio, " es bisiesto"
10         Fin Si
11     Sino
12         Escribir anio, " no es bisiesto"
13     Fin Si
14 FinProceso
```

14.- Realiza un algoritmo que pida al usuario una hora (pedir hora 0-23, minutos 0-59 y segundos 0-59 en elementos separados). A continuación se debe mostrar por pantalla qué hora será pasado un minuto. Utilizar el formato de 24 horas.

```
1  Proceso ejercicio14
2      Definir hora, min, seg como ENTERO
3      Escribir "Introduce hora (0-23)"
4      Leer hora
5      Escribir "Introduce minutos (0-59)"
6      Leer min
7      Escribir "Introduce segundos (0-59)"
8      Leer seg
9      Si min=59 Entonces
10         min<-0
11         Si hora=23 Entonces
12             hora<-0
13         Sino
14             hora<-hora+1
15         Fin Si
16     Sino
17         min<-min+1
18     Fin Si
19     Escribir "Son las ",hora, ":", min, ":", seg
20 FinProceso
```

15.- Realiza la traza de seguimiento del ejercicio anterior. Usa valores extremos, por ejemplo 23 horas 59 minutos 10 segundos, o 0 horas 3 minutos y 59 segundos o las 5 horas 59 minutos 24 segundos

Nº INSTRUCCIÓN	hora	min	seg	ESCRIBE
4	23	-	-	
6	23	59	-	
8	23	59	10	
9,10,11,12,15,18,19	0	0	10	Son las 0:0:10
4	0	-	-	
6	0	3	-	
8	0	3	59	
9,16,17,18,19	0	4	59	Son las 0:4:59
4	5	-	-	
6	5	59	-	
8	5	59	24	
9,10,11,13,14,15,18,19	6	0	24	Son las 6:0:24

16.- Realiza un algoritmo que dé solución al cálculo del factorial de un número entero.

$n! = n \times (n-1) \times (n-2) \times \dots \times 1$

```

1  Proceso ejercicio16
2      Definir n, factorial como ENTERO
3      Escribir "Introduce un número"
4      Leer n
5      factorial<-1
6      Para i<-1 Hasta n Con Paso 1 Hacer
7          factorial<-factorial*i
8      Fin Para
9      Escribir "Su factorial es: ", factorial
10 FinProceso

```

Otra versión con un bucle mientras

```

1  Proceso ejercicio16
2      Definir n, factorial como ENTERO
3      Escribir "Introduce un número"
4      Leer n
5      factorial<-1
6      Mientras n<>0 Hacer
7          ..... factorial<-factorial*n
8          ..... n<-n-1
9      Fin Mientras
10     Escribir "Su factorial es: ", factorial
11 FinProceso

```

17.- Realiza un algoritmo que dé solución al problema de calcular la multiplicación de dos números mediante el uso de sumas sucesivas.

```

1  Proceso ejercicio17
2      Definir n1, n2, producto como enteros
3      Escribir "Introduce el primer número"
4      Leer n1
5      Escribir "Introduce el segundo número"
6      Leer n2
7      producto<-0
8      Para i<-1 Hasta n2 Con Paso 1 Hacer
9          ..... producto<-producto+n1
10     Fin Para
11     Escribir "El producto es: ",producto
12 FinProceso

```

18.- Realiza un algoritmo que pida dos números X e Y de forma que se pretende calcular  $X^Y$  suponiendo que no existe el operador de potencia.

X puede ser cualquier número real distinto de 0

Y puede ser cualquier entero positivo, negativo o nulo.

(Recuerda que  $X^0=1$ ,  $X^{-Y}=1/X^Y$ )

```

1  Proceso ejercicio18
2      Definir base,sol,resultado como real
3      Definir exponente como entero
4      Escribir "Introduce la base"
5      Leer base                      //real distinto de 0
6      Escribir "Introduce el exponente"
7      Leer exponente                //entero positivo, negativo o nulo
8      resultado<-1
9      Si exponente=0 Entonces
10         sol<-resultado
11     Sino
12         //La función ABS devuelve el valor absoluto
13         Para i<-1 Hasta ABS(exponente) Con Paso 1 Hacer
14             resultado<-resultado*base
15         Fin Para
16         Si exponente<0 Entonces
17             sol<-1/resultado
18         Sino
19             sol<-resultado
20         FinSi
21     Fin Si
22     Escribir "Resultado: ", sol
23 FinProceso

```

19.- Algoritmo que lea un valor entero que represente la posición ordinal de un mes (1,2,...12) y nos diga cuantos días tiene ese mes, o un mensaje de error si el valor del mes es incorrecto. Para febrero se indicará 28 o 29 días.

```

1  Proceso ejercicio19
2      Definir mes,numdias como entero
3      Escribir "Introduce ordinal del mes"
4      Leer mes
5      Segun mes Hacer
6          1,3,5,7,8,10,12:
7              Escribir "31 días"
8          4,6,9,11:
9              Escribir "30 días"
10         2:
11             Escribir "28 o 29 días"
12         De Otro Modo:
13             Escribir "Ordinal de mes incorrecto"
14         Fin Segun
15 FinProceso

```

20.- Diseñar un algoritmo para jugar a la adivinanza de números. El ordenador generará un número al azar, entre 0 y 15, y el usuario tendrá que adivinar cuál es. Se dispondrá de todas las oportunidades que quiera y el programa tendrá que ir diciendo si se ha pasado o se quedó corto en cada intento. Finalmente cuando consiga acertarlo se deberá mostrar un mensaje de felicitación y el número de intentos que necesitó. Para generar el número al azar supondremos una función AZAR(x) que genera un número aleatorio entre 0 y x-1.

Como al menos se deberá de repetir una vez el intento de acierto, es más adecuado una estructura REPETIR ..HASTA/MIENTRAS. En cada repetición se pide al usuario el número y se compara con el que hay que acertar, incrementando el número de intentos. Con una estructura SI se comprueba si se pasó, se quedó corto o lo acertó. La condición de salida de REPETIR será que se haya acertado.

```
1  Proceso ejercicio20
2      Definir numsecreto,num,intentos como entero
3      numsecreto<-AZAR(16) //Genera un número entre 0 y 15
4      intentos<-0
5      Repetir
6          Escribir "Qué número crees que es?"
7          Leer num
8          intentos<-intentos+1
9          Si num>numsecreto Entonces
10             Escribir "Te has pasado"
11         Sino
12             Si num<numsecreto Entonces
13                 Escribir "Te has quedado corto"
14             Sino
15                 Escribir "Enhorabuena has acertado. Has necesitado ", intentos, " intentos"
16             Fin Si
17         Fin Si
18     Hasta Que num=numsecreto
19 FinProceso
```