

DAM1

# Practica UD1 Pseudocódigo

Entornos de Desarrollo

Enrique Huerta Lopez  
25-11-2022

# Índice

---

Algoritmo.....	2
Pseudocódigo .....	3
Conclusión .....	4

## Algoritmo

---

En esta tarea vamos a realizar un reloj digital infinito, el cual tendremos que poner en hora.

La lógica es la siguiente:

Cada segundo que pase se sumará 1 al contador de segundos, hasta que su valor sea mayor a 59, en cuyo caso se establecerá a 0 y se le sumará 1 a los minutos. Cuando el valor de los minutos sea mayor a 59, se le sumará 1 a las horas. Por último, cuando el valor de las horas sea igual a 24, se reestablecerá a 0 y vuelta a empezar.

La lógica que hay en este programa viene dada por la instrucción que le dice al sistema que a cada segundo que se suma dura un segundo real, dándonos así el comportamiento real de un reloj.

# Pseudocódigo



```

1 reloj_digital_p1.psc X
2 Algoritmo reloj_digital_p1
3   definir hora,mins,segun2 Como Entero
4   Escribir "Ingresa la hora actual "
5   leer hora
6   Escribir "Ingresa los minutos "
7   leer mins
8   Escribir "Ingresa los segundos"
9   leer segun2
10  //lo primero que hacemos es pedirle al usuario que nos diga desde que hora empezar a contar
11  //como si estuviera poniendo en hora el reloj
12  //Ahora concateno las horas los minutos y los segundos, haciendolo funcionar de la siguiente manera:
13  //Mientras la hora sea menor o igual a 23 (formato de hora 12 horas], minutos menor o igual a 59 y segundos igual,
14  //que lo que haga sea sumar uno
15  Mientras hora ≤ 23 hacer
16      Mientras mins ≤ 59 hacer
17          Mientras segun2 ≤ 59 hacer
18              Limpiar Pantalla //esta linea nos permite que no haga salto de linea cada vez que se cambia el valor,
19              //haciendo que el reloj se quede siempre en el mismo sitio
20              si hora > 9 Entonces //si la hora es mayor que 9 la imprime sin mas
21                  Escribir hora,"h" Sin Saltar
22              SiNo
23                  Escribir "0",hora,"h" Sin Saltar //por el contrario, si la hora es menor a 9, le añade un 0 a la izquierda
24              FinSi
25
26              si mins > 9 Entonces
27                  Escribir ":",mins,"m" Sin Saltar //si los minutos son mayores que 9 lo imprime sin mas
28              SiNo
29                  Escribir ":0",mins,"m" Sin Saltar //por el contrario, si los minutos son menores que 9, le añade un 0 a la izquierda
30              FinSi
31
32              si segun2 > 9 Entonces
33                  Escribir ":",segun2,"s" //si los segundos son menores que 9 los imprime sin mas
34              SiNo
35                  Escribir ":0",segun2,"s" //por el contrario, si los segundos son menores a 9, le añade un 0 a la izquierda
36              FinSi
37              segun2 = segun2+1 //Esta es la clave del codigo, lo que hacemos es sumar 1 a segundos
38              Esperar 1 segundo // cada segundo que pase, ya que esta linea es la que establece el ritmo del programa, tambien por
39              //ello no podemos nombrar a la variable segundos como segundo, ya que son palabras reservadas.
40              //Realmente los segundos son los encargados de hacer que todo funcione y son los unicos que se comportan como un reloj,
41              // ya que tanto los minutos como las horas lo unico que hacen es esperar a que los segundos terminen por darles el valor.
42          FinMientras
43          segun2 = 0 //cuando los segundos superan 59, los reestablece a 0
44          mins = mins +1 //y le suma 1 a los minutos
45      FinMientras
46
47      mins = 0 //cuando los minutos superan 59, los reestablece a 0
48      hora = hora +1 // y le suma 1 a las horas
49      si hora == 24 entonces //cuando la hora es igual a 24
50          hora = 0 //reestablece la hora a 0 y vuelve a empezar a contar, por lo que es infinito
51      FinSi
52  FinMientras
53 FinAlgoritmo //Final del programa

```



## Conclusión

---

Al realizar este ejercicio he entendido mejor como funcionan los condicionales, así como su sintaxis en Pseint, la cual hasta ahora me resultaba confusa.

Una actividad a priori compleja, pero que cuando te pones a hacerla te das cuenta de que no lo es tanto.