

Sumario

UT 03: Elementos de un programa informático.....	2
1 Ejercicios en pseudocódigo (PSEInt).....	2
1.1 Exponente.....	2
1.2 Serie de Fibonacci.....	2
1.3 Área de un círculo.....	2
1.4 Factorial.....	2
1.5 Números primos.....	2
1.6 Conversor binario.....	3
1.7 Número de cifras.....	3
1.8 Pirámide.....	3
1.9 Longitud de una cadena.....	3
1.10 Array.....	3
1.11 Array con valores aleatorios.....	3
1.12 Suma array.....	3
1.13 Media aritmética.....	4
1.14 Adivinar número.....	4
2 Crear diagramas de flujo.....	4
3 Crear programas a partir del pseudocódigo.....	4

UT 03: Elementos de un programa informático

1 Ejercicios en pseudocódigo (PSEInt)

1.1 Exponente

Escribe una función en pseudocódigo que devuelva el resultado de un número elevado a un exponente dado.

1.2 Serie de Fibonacci

Escribe una función en pseudocódigo que devuelva el término N (siendo N un número entero positivo) de la serie de Fibonacci, esta sigue la siguiente serie: 1, 1, 2, 3, 5, 8, 13, 21... y así sucesivamente.

Para obtener un número se suman los dos números anteriores. Por ejemplo, si introducimos N=3, la función nos devuelve el 2 (el tercer elemento de la serie).

1.3 Área de un círculo

Escribe una función en pseudocódigo que devuelva el área de un círculo dado su radio. Recuerda que para calcular el área, la fórmula es $\pi * r^2$

1.4 Factorial

Escribe una función en pseudocódigo que devuelva el factorial de un número N (siendo N un número entero positivo).

Haz después un algoritmo principal que pida por teclado dos números y escriba en pantalla los factoriales de cada número comprendido entre los dos anteriores. El factorial de 5 es el resultado de $5*4*3*2*1$

1.5 Números primos

Escribe una función en pseudocódigo que devuelva si un número dado es primo o no. Para que un número sea primo solo es divisible entre 1 y sí mismo (por ejemplo, 13, 17, 19...).

Utilizando la función, escribe un programa que escriba todos los números primos entre 1 y un número leído desde el teclado. Piensa qué debe devolver la función.

1.6 Conversor binario

Escribe una función en pseudocódigo que dado un número decimal (en base 10), entre 0 y 15, devuelva el valor equivalente en sistema de numeración binario(en este caso, un numero binario de 4 dígitos). Es aconsejable que los dígitos binarios sean una cadena, para poder concatenar entre sí. Por ejemplo, si introducimos un 7, nos devolverá 0111. Si se introduce un numero menor que 0 o mayor que 16, nos mostrara un mensaje de error.

1.7 Número de cifras

Algoritmo que, dado un número entero (este numero no podrá ser menor o igual que 0), determine el número de cifras que tiene. Por ejemplo, para un 253, devuelve un 3.

1.8 Pirámide

Escribe una función en pseudocódigo que dibuje una pirámide invertida en pantalla como la de la figura. La altura se pasará como parámetro. Si se pasa una altura =0 o negativa, la función devolverá un error.

Ejemplo para altura = 5

```
*****
*****
*****
***
*
```

1.9 Longitud de una cadena

Crear un subproceso que devuelva la longitud de una cadena.

1.10 Array

Crear un array con una serie de valores y utilizar una función para que los muestre todos correlativamente.

1.11 Array con valores aleatorios

Crear un subproceso rellene un array con números aleatorios, pasándole un array y los números entre los que estarán los valores.

1.12 Suma array

Crear un subproceso que devuelva la suma de un array pasado por parámetro.

1.13 Media aritmética

Crear un subproceso que devuelva la media de un array pasado por parámetro.

1.14 Adivinar número

Vamos a realizar el pequeño juego de adivinar un número entre 1 y 100.

Tendremos que programar los siguientes subprocesos:

- `LeerNumero()`: Pide un numero y hasta que el usuario no escribe un valor entre 1 y 100, vuelve a pedir el valor.
- `comprobarValor(numeroUsuario, numeroCorrecto)`: comprueba si el numero es correcto, este devuelve un numero que puede ser: 0: los dos numeros son iguales 1: el numeroUsuario es mayor que el numeroCorrecto -1: el numeroUsuario es menor que el numeroCorrecto

2 Crear diagramas de flujo

Para los ejercicios anteriores, crear los correspondientes diagramas de flujo mediante la herramienta PseInt.

3 Crear programas a partir del pseudocódigo

Para los ejercicios anteriores, realizar la conversión del pseudocódigo en código C. Compilar y ejecutar en un sistema Linux.

Posteriormente realizar la misma operativa en código Java.