

EXAMEN 1ª Eval. DAW 1. PROGRAMACIÓN

10 de diciembre – Turno A

Nota: crea una carpeta que contendrá todo “el contenido de tu examen”. Dicha carpeta será recogida por tu profesor al finalizar. Los ejercicios se pueden realizar en cualquier orden. De cada ejercicio debes entregar al menos su código de forma legible y entendible en “una versión” (se valorará el uso de comentarios). **Para ello entregarás como mínimo, de cada ejercicio, un archivo con la extensión “.txt”**. Además puedes entregar “la carpeta o archivos que genere” (sin dejarte nada para su correcto funcionamiento) del IDE con el que has llevado a cabo la programación.

1. Hacer un programa que pida los tipos primitivos por consola y los muestre por ventana. **(0,5 puntos)**.
2. Hacer un programa que pida “un número de mes” y “un número de día” y nos escriba si se trata de una fecha correcta o no, si lo fuera mostrará el “nombre del mes” y la cantidad de días totales que tiene. Si no lo fuera deberá mostrar en qué falla, si el número de mes, número de día, o ambas cosas (febrero consideraremos que tiene como mucho 28 días). **(1,5 puntos)**.
3. Hacer un programa que pida números enteros, hasta que se introduzca un número más pequeño que -30. Nos deberá mostrar la cantidad de ceros introducidos, la cantidad de negativos introducidos, la cantidad de positivos introducidos. La media de todos ellos. El mayor y el menor. **(1,5 puntos)**.
3. Hacer un programa que pida al usuario una cantidad de dinero en euros (se debe admitir hasta dos decimales) y devuelva dicha cantidad en los tipos de monedas existentes en la actualidad, es decir (de 2 euros, de 1 euro, de 50 céntimos, de 20 céntimos, de 10 céntimos, de 5 céntimos, de 2 céntimos y de 1 céntimo. (Pista, la idea que debemos emplear es prácticamente la misma que utilizamos en nuestro ejercicio de clase de dada una cantidad en segundos, mostrar su valor en días, horas, minutos, segundos...) **(2 puntos)**.
4. Hacer un programa que genere de forma automática un número secreto del 1 al 20. Al usuario se le darán 8 intentos para adivinarlo, ayudándolo en cada intento, diciendo si el número secreto es mayor o menor que el que ha introducido. El programa termina si se ha llegado al tope de intentos (indicando que ha perdido), o bien si adivina el número (indicando que ha ganado y el número de intentos). **(2 puntos)**.
5. **(2,5 puntos)** Hacer un programa que pida un entero (este deberá ser positivo, impar, menor que 15, si no fuera así, deberá insistir hasta que se introduzca un valor válido). Una vez introducido un valor válido dibujará un triángulo de la siguiente forma:

Si el valor pasado es 7:

4	1 2 3 *	4	1	1	—	—	—
3	1 2 * * *	3	2	3			
2	1 * * * * *	2	3	5			
1	* * * * * *	1	4	7			

fila 1

// en la base tiene tantos asteriscos como indica el número entero que pedimos.

→ *matres set hasta indexical.*

→ *Readline stocavstre, es cueva*

(0,00 ~ 0,2]

200	0,20
- 80	- 0,8