

T2. Introducción al lenguaje JavaScript

Basados en los ejercicios de C de mi compañera de M^a Asunción Criado.

BOLETÍN DE EJERCICIOS

ESTRUCTURA CONDICIONAL

1. Calcular el área¹ y el volumen de la esfera cuyo radio se pide al usuario².
2. Diseñar un algoritmo para hallar el valor absoluto de un número leído de teclado y presentarlo en pantalla.
3. Realiza un algoritmo que lea un dato de teclado y calcule e imprima su inverso. Considere el caso especial del valor 0 mostrando el mensaje de error correspondiente.
4. Escriba un algoritmo que lea un instante de tiempo dado en horas y minutos y escriba como respuesta los mensajes “Buenos días” “Buenas tardes” “Buenas noches”, de acuerdo con las siguientes reglas:
 - a. Es de día desde las 7:30 hasta las 14:00 horas.
 - b. Es tarde desde las 14:01 hasta las horas 20:30.
 - c. Es noche desde las 20:31 hasta las horas 7:29.
5. Para comprobar si un año es o no bisiesto se usa la siguiente regla: “Un año es bisiesto si es divisible por 400, o bien es divisible por 4 pero no por 100”. Diseñar un programa que utilizando una variable lógica que tenga valor cierto si el año es bisiesto y falso si no lo es.
6. Dado un CAPITAL un RÉDITO determinado y un TIEMPO de imposición, calcular el **INTERÉS simple**³ que se produce.
7. Diseñar un algoritmo que tomando las coordenadas cartesianas de un punto en el plano y calcule e imprima el cuadrante al que pertenece dicho punto. También debe detectar cuando está en el origen o sobre un eje.
8. Realizar un programa que calcule las soluciones de una ecuación de segundo grado. Se debe contemplar que a veces no tiene solución y notificarlo al usuario.

¹ El número π se obtiene con la función **Math.PI**

² La función **prompt(cadena_de_texto)** devuelve un valor solicitado al usuario mostrando el texto *cadena_de_texto*.

³ Google es tu amigo.

9. Elabora un programa que dados un precio y una cantidad pagada, obtengamos el cambio con el mínimo número de elementos (monedas o billetes). Para devolver el cambio disponemos de una cantidad ilimitada de monedas y de billetes.

ESTRUCTURAS ITERATIVAS Y SELECCIÓN MÚLTIPLE

1. Diseñar un algoritmo para escribir en pantalla los n primeros números naturales, sus cuadrados, sus cubos y la suma de todos ellos.
2. Hallar el menor, el mayor y la media de un conjunto de números positivos leídos de teclado.⁴
3. Dada la función $y = 1/4 x$ diseñar un programa que para cada valor de abscisa leído por teclado dé el correspondiente a la ordenada.
4. Realizar un programa que proporcione el cambio de Euros a Dólares, Libras, Yenes, Franco Suizo, Corona Noruega, según opción. El programa debe controlar todas las entradas y ofrecer al usuario la posibilidad de repetir o salir.
5. Hacer un algoritmo que lea el número correspondiente a un mes del calendario y presente en pantalla su nombre usando una estructura de control adecuada.
6. Realizar un algoritmo que permita introducir la nota de una asignatura por teclado, la valide para que esté comprendida entre 0 y 10 y se escriba en letras de la siguiente manera:
SUSPENSO si es menor que 5
APROBADO mayor que 5 y menor que 6
BIEN entre 6 y 7
NOTABLE entre 7 y 8
SOBRESALIENTE entre 9 y 10
7. Realizar un programa que lea una fecha de nacimiento de la forma día, mes, año, y dé como resultado el número de Tarot. El programa verificará si la fecha es correcta. El número de Tarot se haya sumando los números de la fecha de nacimiento y reduciéndolos a un único dígito. Por ejemplo, si su fecha de nacimiento es 20 de julio de 1984, el número de Tarot sería: $20 + 7 + 1984 = 2011 \Rightarrow 2 + 0 + 1 + 1 = 4$

TODAS LAS ESTRUCTURAS DE CONTROL

NOTA: Validar las entradas oportunas y construir los algoritmos de manera que se ejecuten cuantas veces deseemos con el fin de depurar el código sin salir del programa.

⁴ Usar la función **parseInt**(texto) para convertir a número la entrada del usuario.

8. Estamos interesados en calcular los ingresos medios de un conjunto de hombres y mujeres. Para ello disponemos de un documento donde se recoge si se trata de un hombre (H) o una mujer (M) y su sueldo correspondiente. La entrada de datos termina cuando se lee un * como sexo. Se validarán todas las entradas, el sexo será H o M y el sueldo entre los 1000 y 2000 euros independientemente de que el trabajador sea hombre o mujer.
9. Diseñar un programa que, dada una cierta cantidad de dinero que se leerá desde la entrada estándar, calcule cuál es el número mínimo de monedas de curso legal que equivalen a dicha cantidad de dinero.
10. Diseñar un programa que lea de teclado un número entero positivo en base 10 y escriba su correspondiente representación binaria usando la técnica de divisiones sucesivas.
11. Realiza un algoritmo que calcule la posición relativa de dos circunferencias. Para ello el programa debe leer los centros (x,y) y los radios de ambas. La información que necesitas referente al tema en cuestión, la encontrarás en la dirección electrónica que aparece a continuación:
<http://concurso.cnice.mec.es/cnice2006/material098/geometria/geoweb/circun3.htm>
12. Escribe un algoritmo que lea de teclado las coordenadas de cuatro puntos (x,y) en el plano e indique si estos puntos forman un cuadrado o un rectángulo. En ambos casos deberá calcular el área y el perímetro.
13. Escribe un algoritmo que lea desde la entrada estándar dos fechas dadas por día, mes y año y calcule cuál de ellas es anterior a la otra.
14. Realiza un algoritmo que solicite del usuario un tiempo dado en segundos y calcule y presente en pantalla dicho tiempo pero expresado en horas, minutos y segundos.
15. Diseña un algoritmo para simular el juego de ¿dónde está la bolita? (trile), famoso timo practicado por los llamados trileros .En nuestro caso el ordenador será un honrado trilero que no engañará al usuario que juegue con él.
16. Realiza un algoritmo en JavaScript que calcule la posición relativa de dos rectángulos.

Para ello el programa debe leer los vértices inferior izquierdo(a) y superior derecho de cada rectángulo(b). En el diagrama estos puntos son las parejas

$$\{ A(x_a, y_a), B(x_b, y_b) \} \text{ y } \{ C(x_c, y_c), D(x_d, y_d) \}$$

EJERCICIO 1. Realiza un algoritmo en JavaScript que calcule la posición relativa de dos rectángulos.

Para ello el programa debe leer los vértices inferior izquierdo(a) y superior derecho de cada rectángulo(b). En el diagrama estos puntos son las parejas

$$\{ A(x_a, y_a), B(x_b, y_b) \} \text{ y } \{ C(x_c, y_c), D(x_d, y_d) \}$$



Especificaciones / restricciones:

- El punto A siempre será menor que el punto B.
- A su vez, el punto C siempre será menor que el punto D.
- Por último, el punto C será mayor o igual al A.
- Los rectángulos deben tener área positiva y mayor que 0. (Área = base * altura)
- Validar lo anterior y las entradas oportunas y construir el algoritmo de manera que se ejecute cuantas veces deseemos.
- Todos los casos deben estar contemplados: contenidos, idénticos, se cortan y exteriores.
- Utilizar las estructuras de control que más se adecuen.
- Utilizar código autodocumentado: Los subprogramas y las variables deben tener nombres alusivos.
- De existir errores de ejecución deben ser simples.



Especificaciones / restricciones:

- El punto A siempre será menor que el punto B.
- A su vez, el punto C siempre será menor que el punto D.
- Por último, el punto C será mayor o igual al A.
- Los rectángulos deben tener área positiva y mayor que 0. ($\text{Área} = \text{base} * \text{altura}$)
- Validar lo anterior y las entradas oportunas y construir el algoritmo de manera que se ejecute cuantas veces deseemos.
- Todos los casos deben estar contemplados: contenidos, idénticos, se cortan y exteriores.
- Utilizar las estructuras de control que más se adecuen.
- Utilizar código autodocumentado: Los subprogramas y las variables deben tener nombres alusivos.
- De existir errores de ejecución deben ser simples.

17.

DEPURACIÓN DE ERRORES

18. Usa cualquiera de los programas que has codificado en JavaScript e incluye errores deliberadamente. La lista de errores que se sugiere son los siguientes:
- a. Omite uno de los paréntesis de alguna estructura.
 - b. Omite los dos paréntesis de alguna estructura.
 - c. Escribe mal alguna de las palabras reservadas de JavaScript.
 - d. No escribas la llave de cierre de alguna estructura.
 - e. Escribe mal el nombre de algunas de las variables declaradas.
 - f. etc