

EJERCICIOS DE BUCLES

ENTREGABLES

- 1) Implementa un programa que calcule y muestre la suma de los números enteros entre 1 y 50.
- 2) Implementa un programa que lea un carácter hasta que el usuario introduzca un carácter mayor que 'A' y menor que 'Z'.
- 3) Realiza un programa que solicite al usuario un año de nacimiento y muestre a continuación en pantalla el signo que le corresponde según el horóscopo chino imaginario: si la suma de las cifras del año es par, el elemento es el fuego. En caso contrario, es el agua. PISTA: Puedes usar los operadores % y / para poder acceder a las distintas cifras del número.

A REALIZAR DURANTE LAS SESIONES DE PRACTICAS Y EN CASA

Problema 1: Regalando caramelos

La empresa Dulcemelos ha decidido emprender una estrategia innovadora de protección del medio ambiente a la par que alegra a los niños. Para ello, el día del caramelo ha decidido regalar caramelos a aquellos niños que lleven los envoltorios de los caramelos ya consumidos, de esta forma los niños no tiran los envoltorios al suelo. Eso sí, sólo se permite acudir una vez a la empresa a canjear los envoltorios. Por cada 5 envoltorios, regala un caramelo. Eso significa que si un niño compró 10 caramelos, podrá en realidad disfrutar de 12 caramelos: al comerse los 10 caramelos tendrá 2 caramelos más gratis. Para que sea rentable, a la empresa le interesa que a los niños les queden envoltorios no canjeables.

Entrada

Una serie de números que son los números de caramelos que compran los niños. El 0 indica que ya no hay más datos

Salida

Número de caramelos vendidos, número de caramelos regalados y mensaje indicando si es rentable o no.

Casos de prueba

Entrada	Salida
12 25 31 0	68 13 RENTABLE
14 14 14 14 0	56 8 RENTABLE
25 25 5 0	55 11 NO RENTABLE
24 24 4 3 0	55 8 RENTABLE

Problema 2: Códigos de barras

En 1952, se diseñó el famoso sistema de los códigos de barras. Consiste en una técnica para representar números (y a veces letras) mediante una serie de líneas verticales paralelas, con diferentes grosores y separaciones entre ellas. La manera concreta de codificar mediante barras los números y las letras puede ser muy variada, lo que ha llevado a la aparición de diferentes estándares. De todos ellos, el EAN (*European Article Number*) resulta ser el más extendido. Existe así el llamado EAN-8, que codifica con 8 dígitos.



El último dígito del código se utiliza para detección de errores, y se calcula a partir de los demás. Para eso:

- Empezando por la derecha (sin contar el dígito de control que se está calculando), se suman los dígitos individuales, multiplicados por un factor:
 - Los dígitos en posiciones impares (empezando a contar por la derecha saltándonos el de control) se multiplican por 3.
 - Los dígitos en posiciones pares se multiplican por 1.

Por ejemplo, para el código EAN-8 de la figura la operación a realizar es:

$$2 \cdot 3 + 5 \cdot 1 + 9 \cdot 3 + 3 \cdot 1 + 8 \cdot 3 + 5 \cdot 1 + 6 \cdot 3 = 88$$

- El dígito de comprobación es el número que hay que sumar al resultado anterior para llegar a un valor múltiplo de 10. En el ejemplo de EAN-8, para llegar al múltiplo de 10 más cercano por encima del número 88 hay que sumar 2 (y llegar al 90). Ten en cuenta que si la suma resulta ser ya múltiplo de 10, el dígito de control será 0.

Entrada

La entrada estará formada por un número de 8 dígitos

Salida

Si el dígito de control es correcto escribirá "SI". En otro caso, escribirá "NO".

Casos de prueba

Entrada	Salida
65839522	SI
65839529	NO
23467629	NO

Problema 3: El e-tutor

En el colegio nos han pedido que desarrollemos un programa para ayudar a los niños a aprender a multiplicar. Para ello, el programa debe tener el siguiente funcionamiento:

Para ello:

- Generar dos números aleatorios n1 y n2 entre 0 y 9
- Pedir el resultado de la multiplicación al usuario.
- Si el resultado es correcto aumentar los aciertos y mostrar que ha acertado. Si no ha acertado, indicarle cómo es el resultado con respecto al número que el usuario ha introducido: mayor o menor. Se le da otra oportunidad de acertar la multiplicación.
- Se le preguntan 5 multiplicaciones distintas y al final se le otorga una puntuación. Un acierto a la primera es un punto, un acierto con dos oportunidades en medio punto.

Además, al finalizar las 5 multiplicaciones, se le pregunta al usuario si quiere volver a jugar y si es así se repite el juego desde el principio.

Entrada

El resultado de la multiplicación de los dos números aleatorios que genera el ordenador. Hay 5 multiplicaciones distintas y la respuesta (s/n) a la pregunta de si quiere repetir el juego

Salida

La palabra Acierto o Fallo para cada suma y la puntuación conseguida

Ejemplo de ejecución

```
3*6= 18
Acierto!
1*4= 4
Acierto!
4*8= 31
Error!: LA SOLUCION ES MAYOR
4*8= 32
AHORA HAS ACERTADO!!
1*5= 5
Acierto!
9*1= 9
Acierto!
PUNTUACION: 4.5
Quieres jugar otra vez(s/n)? _
```

ANEXO: GENERAR NÚMEROS ALEATORIOS

```
#include <stdlib.h> //se debe añadir esta librería
...
int i,n;
srand((unsigned)time(NULL)); //semilla
for (i=0;i<10;i++)
{
    n=rand()%10; //genera un número entre 0 y 9
}
```

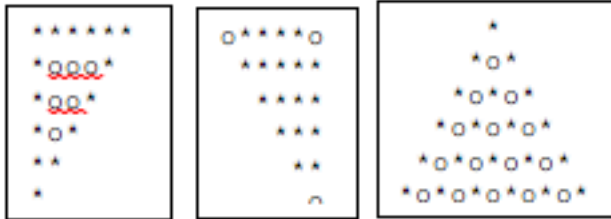
```

        cout << n<<endl;
    }

```

Problema 4: Dibujando

Implementa un programa que visualice en pantalla un menú que permita elegir entre las siguientes figuras para dibujar preguntando al usuario el número de filas n que deben tener las figuras (En el ejemplo mostrado, $n = 6$). Ten en cuenta que la solución que propongamos debe contener sentencias de salida por pantalla en las que se impriman exclusivamente uno solo de los siguientes caracteres: blanco ' ', asterisco '*' y letra 'o'.



Entrada

la opción del menú y un número que indica el número de filas de los triángulos

Salida

el triángulo elegido