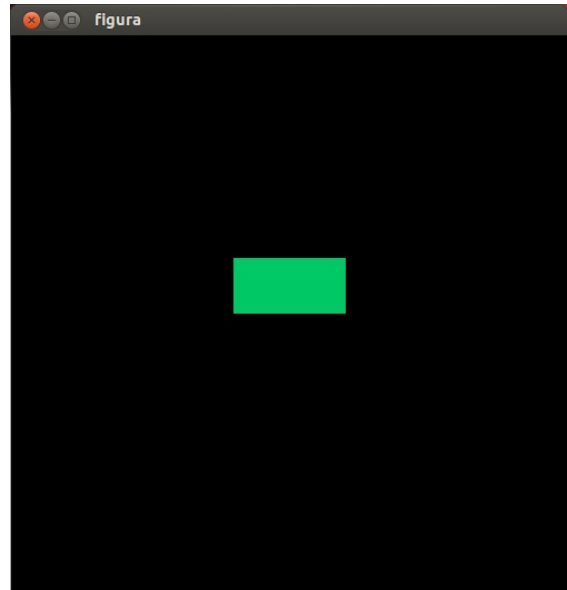
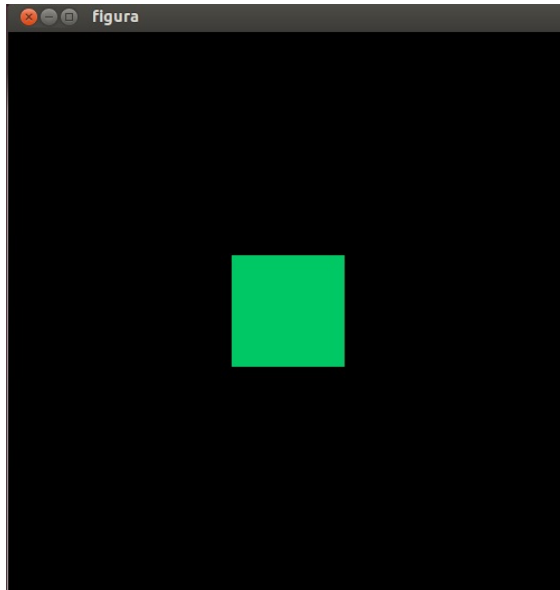


EJERCICIOS DE BUCLES

A REALIZAR DURANTE LAS SESIONES DE PRACTICAS Y EN CASA (3 oct-16 oct)

Problema 1: Dibujando figuras densas

Basándote en el problema 1 de la práctica 3 (dibujar un cuadrado o un rectángulo), escribe un programa que dibuje el cuadrado o el rectángulo relleno empleando la librería gráfica gfx.



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)?
```

¿CÓMO GENERAR NÚMEROS ALEATORIOS?

Este ejemplo sirve para generar 2 números aleatorios con valores entre 0 y 9

```
#include <stdlib.h> //se debe añadir esta librería
#include <time.h>    //se debe añadir esta librería

...
int n1, n2;
srand(time(NULL)); //semilla
n1=rand()%10; //genera un número entre 0 y 9
n2=rand()%10; //genera un número entre 0 y 9

cout << n1 << " " << n2 << endl;
...
```

Problema 4: Ayudando al agente 00Negativo

Nos han encargado ayudar al agente 00Negativo a decodificar sus mensajes. El mensaje que tiene que recibir consiste en un número formado por varias cifras, pero el número real es un número de una

sola cifra. Para decodificar el número hay que sumar todas sus cifras repetidamente hasta que quede un número de una sola cifra

Entrada

Un número entero formado por varias cifras

Salida

Un número entero de una sola cifra

Casos de prueba

Entrada	Salida
45678	3
321	6
19978	7
8	8