

EJERCICIOS DE PROGRAMACION MODULAR

Problema 1: El conversor de temperaturas

Realiza un programa en C que permita convertir grados Celsius a Fahrenheit y viceversa. El programa debe mostrar un menú para poder seleccionar qué opción se desea (más la opción terminar). Y preguntar la temperatura que desea convertir. **Cada una de las conversiones se debe realizar por medio de un módulo.**

- La conversión de grados Celsius a grados Fahrenheit se obtiene multiplicando la temperatura en Celsius por 1,8 y sumando 32.
- La conversión de grados Fahrenheit a grados Celsius se obtiene restándole 32 a la temperatura en grados Fahrenheit y dividiéndolo por 1,8.

Ejemplo de ejecución:

Seleccione qué tipo de conversión desea realizar:

1. Convertir grados Celsius a Fahrenheit.
2. Convertir grados Fahrenheit a Celsius.
3. Terminar el programa

Opción:1

Introduzca la temperatura en grados Celsius: 30

La temperatura en grados Fahrenheit es: 86

Problema 2: El juego del Rojo-Amarillo-Verde

Escribir un programa que implemente el juego del Rojo-Amarillo-Verde. El programa genera tres números aleatorios distintos entre 0 y 9. El objetivo del juego es adivinar los números así como sus posiciones correctas con el menor número de intentos. Para cada intento, el usuario proporciona 3 números. El programa responde con una pista que puede ser rojo, amarillo o verde. Si un número está en la posición correcta, la respuesta es verde. Si el número está en una posición incorrecta la respuesta es amarillo. Si el número no coincide con ninguno de los tres números generados por el ordenador, la respuesta es rojo.

Detalles de implementación

Además del main(), debe haber varios módulos:

- Módulo que se encarga de generar los tres números aleatorios. Este módulo debe devolver al main() los números generados por el ordenador. Este módulo debe asegurarse de que los números generados son distintos entre si antes de enviarlos al main().

- Módulo/s para pedir los números al usuario. Este módulo debe controlar que los números solicitados estén entre 0 y 9 y que sean distintos entre si. Este módulo debe devolver los números al `main()`.
- El `main()` se ocupa de realizar las llamadas a los módulos correspondientes.

Ejemplo de ejecución:

Introduce un número: 4

Introduce un número: 7

Introduce un número: 0

4 7 0 Rojo Rojo Rojo

Introduce un número: 12

El número debe estar entre 0 y 9

Introduce un número: 1

Introduce un número: 5

Introduce un número: 9

1 5 9 Rojo Amarillo Verde

Introduce un número: 5

Introduce un número: 2

Introduce un número: 9

5 2 9 Verde Rojo Verde

Introduce un número: 5

Introduce un número: 3

Introduce un número: 9

5 3 9 Verde Verde Verde

FIN DEL JUEGO

Problema 3: El e-Tutor 2.0

En el colegio han quedado encantados con el programa que hicimos para ayudar a los niños a aprender a multiplicar, así que nos han pedido que lo mejoremos para ayudarles a practicar y aprender con todas las operaciones. En este caso, el programa debe generar 7 operaciones consecutivas que el estudiante debe ir respondiendo correctamente para ir avanzando. Los números aleatorios que genera el ordenador deben estar entre 1 y 9 y las operaciones aritméticas a considerar son la suma, la resta, la multiplicación y la división. El resultado de una operación (si es correcta) se convierte en el primer operando de la siguiente operación. La instrucción termina cuando el estudiante ha acertado 7 operaciones. Cuando falla en una operación, se vuelve a generar otro operador y otro valor para el segundo operando. El diseño del programa debe ser modular.

Ejemplo de ejecución:

```
3-3= 0
¡ ¡HAS ACERTADO!!
0+1= 1
¡ ¡HAS ACERTADO!!
1*6= 7
Ups. Vuelve a intentarlo
7/6= 1
¡ ¡HAS ACERTADO!!
1/1= 1
¡ ¡HAS ACERTADO!!
1-2= -1
¡ ¡HAS ACERTADO!!
-1*1= -1
¡ ¡HAS ACERTADO!!
-1*4= -4
¡ ¡HAS ACERTADO!!
```