

Examen parcial de prácticas

Se valorará:

- El uso correcto de los módulos propuestos.
- El uso de comentarios que describan lo que se está realizando.
- La correcta organización del código (sangrado).
- **NO SE DEBEN** utilizar variables globales.

Debes implementar un programa en C que permita jugar al juego de apuestas Pares o Nones. El usuario empezará el juego con 100 fichas e irá apostando a pares o nones. Si gana doblará las fichas apostadas, si pierde perderá lo que ha apostado. El juego terminará cuando el usuario ya no quiera seguir jugando o cuando se quede sin fichas. Además el programa deberá almacenar y mostrar el número de veces que ha ganado el usuario.

Módulos:

Para implementar el juego, además de la función `main()`, el programa debe contener los siguientes módulos:

- Módulo “cuantoApostar”. Este módulo le mostrará al usuario las fichas que tiene y le pedirá la cantidad de fichas que quiere apostar. Se tendrá en cuenta si se comprueba si esa cantidad es correcta. El módulo **debe devolver la cantidad** que el usuario quiere apostar.
- Módulo “paresOnones”. Este módulo debe pedir al usuario si quiere apostar por Pares o por Nones. Se tendrá en cuenta si comprueba que la elección del usuario es correcta. El módulo debe mostrar la apuesta del usuario y **debe devolver la elección** del usuario (Pares (p) o Nones (n)).
- Módulo “lanzarDedos”. Este módulo mostrará el total de dedos y devolverá si ha salido pares o nones. Debe calcularlo de manera aleatoria, para ello debe utilizar la función `rand()`, que genera un número aleatorio. Ejemplo:

```
valor = rand() % 5; //porque son 5 los dedos de una mano. Para utilizar la función
rand() para generar un número aleatorio debemos incluir en nuestro programa la
librería: #include<stdlib>
```

Y para que en cada ejecución los valores sean diferentes debes escribir al inicio:
`srand(time(NULL));`

- Módulo “comprobarApuesta”. Este módulo recibirá todos los parámetros necesarios para calcular si el usuario ha ganado la apuesta o no. Comprobará:
 - Si los “dedos” que se muestran corresponde con la apuesta del usuario, ha ganado y por tanto a las fichas que tenía se le sumarán la apuesta * 2.
 - Si el usuario ha perdido se le restará la apuesta de sus fichas.

El módulo mostrará por pantalla si el usuario ha ganado o perdido.

Apellidos y nombre:

Ejemplo de ejecución básico:

```
*****
***** REPASANDO MATES *****
*****
Tienes 100 fichas, ¿cuánto quieres apostar? 50
¿Pares (p) o nones (n)? p
Apuesta realizada. Has apostado 50 fichas a PARES.
 /|  /|  /|  /|  /|
Han salido NONES
Has fallado. Lo siento
Has ganado 0 veces.

¿Quieres jugar otra vez? s
Tienes 50 fichas, ¿cuánto quieres apostar? 50
No puedes apostar 50, solo tienes 50 fichas.
¿Cuánto quieres apostar? 40
¿Pares (p) o nones (n)? p
Apuesta realizada. Has apostado 40 fichas a PARES.
 /|  /|  /|  /|
Han salido PARES
¡Has acertado!
Has ganado 1 veces.

¿Quieres jugar otra vez? s
Tienes 130 fichas, ¿cuánto quieres apostar? 130
¿Pares (p) o nones (n)? n
Apuesta realizada. Has apostado 130 fichas a NONES.
 /|  /|  /|
Han salido NONES
¡Has acertado!
Has ganado 2 veces.

¿Quieres jugar otra vez? s
Tienes 390 fichas, ¿cuánto quieres apostar? 390
¿Pares (p) o nones (n)? p
Apuesta realizada. Has apostado 390 fichas a PARES.
 /|
Han salido NONES
Has fallado. Lo siento
Has ganado 2 veces.

¿Quieres jugar otra vez? s
Lo siento, no te quedan fichas para continuar
HASTA LA PRÓXIMA
```