

Nom i Cognom:

Receta de la familia (2,75 pts)

Encuentras en el desván una libreta de recetas de tu bisabuela de 1832, para un pastel que dice "Receta especiales de la familia XXXX (aquí tu apellido)", quieres honrar a tu abuela. Empiezas por las que tienen chocolate.

Pero todas las recetas de tu abuela, la cantidad la expone como **cuadraditos**, Supones que cuadraditos no han cambiado y son esto...



Realiza un programa que te permita saber cuántas tabletas de chocolate tienes que comprar para cada receta.

El programa te **pedirá el número de cuadraditos** que pone en la receta y luego te pedirá las **filas y columnas** de cuadraditos de la tableta que vas a comprar en el super. Y el programa te dirá cuantas tabletas mínimas tienes que comprar.

Después de calcular te pedirá si quieres volver a calcular otra receta, si pones S vuelves a empezar, cuando pongas N termina el programa y debe decir cuantas recetas has calculado. **Recordad fflush(stdin).**

El programa tendrá dos funciones.

-Cuadraditos_por_tableta (0,5 pts)

/*Recibirá de entrada dos parámetros
filas: filas que tiene la tableta que compras.
columnas: columnas que tiene la tableta que compras
Devuelve el **número de cuadraditos** que tiene la tabla*/

-Tabletas_comprar (0,75 pts)

/* Recibirá de entrada dos parámetros
cuadraditos_receta: Los cuadraditos que pone en la receta
cuadraditos_tableta: Número de cuadraditos que tiene tu tableta
Devuelve el número de tabletas mínimo que debes comprar*/

-Estructura general programa, recoger bien los datos(1,25), preguntar si quieres volver a calcular otra receta y decirte cuantas recetas has calculado

Conquistando bases imperiales (3 puntos)

Para su próxima ofensiva, el ejército rebelde ha decidido conquistar varias de las bases enemigas cercanas al planeta Polis Massa. Pero no saben si las podrán conquistar todas o no con los soldados que tienen. El número de bajas es aleatoria entre 1 y 20. Además, se espera que las fuerzas imperiales no se queden con los brazos cruzados ante cada conquista, por lo que es necesario dejar **10 soldados de retén militar siempre** en las bases conquistadas.

Necesitamos saber si podemos conquistarlas todas.



El programa debe pedir cuantos soldados tienes.

También cuantas bases quieres conquistar (prueba con pocas inicialmente) , un número N.

Y luego para cada base irá pidiendo cuantas bajas hay en esa conquista.

El programa te irá pidiendo N veces, las bajas de cada base, si en algun momento, ya no tienes soldados deberá terminarse, y si al final las has conquistado todas, al final te dirá cuantos soldados te han sobrado para ir a celebrarlo.

Soldados_mueren(0,5 pts)

Calcula cuantos soldados pereceran. Son bajas aleatorias pero mínimo muere 1 y máximo 20

Parametros entrada : ninguno

Parametros salida: Número de bajas

Recomendación para probar si va bien aqui dentro imprime cuantas bajas os da...

Luego ya quitaras el printf

Soldados_restantes(0,75-1,25)

Recibirá **los soldados que aún tienes**,

Recibirá **los soldados que mueren en esa base**

Recibirá **los soldados de retén , ese número es fijo según enunciado.**

Devolverá los soldados que aun te quedan, nunca podrá devolver datos incoherentes.

Para que valga 1,25 , el **primer parámetro recibido se pasará por referencia** y por tanto al salir este mismo parámetro debe tener los soldados restantes.

Si no sale por referencia valdrá 0,75 y **la función devuelve con un return** el número de soldados restantes.

Programa completo (1,25 pts)

Debe pedir las bases que quieres conquistar y los soldados que tienes

Debe ir pidiendo las bajas hasta saber si no las has podido conquistar todas o si

Mostrar cuantos soldados aún tienes si has conseguido ganar.

Ejercicio 3 recursividad (2,75 pts)

Escribir un programa que encuentre la suma de los enteros positivos pares desde N hasta 2. Chequear que si N es impar se imprima un mensaje de error y no calcule nada.

- Si ponen por ejemplo el número 7, debe **devolver un valor inequívoco** para que el main de un mensaje de error (**0,25 pts**).
- Si ponen el número 18, tiene que sumar el 18, el 16, el 14, hasta el último número par (**acertar caso base 0,5 pts, caso recursivo 1,75 pts**).

Ejercicio 4 tragaperras (1,75 pts)

El mecanismo de una máquina tragaperras tradicional es fácil. Se dispone de 3 ruletas con diferentes símbolos, en cada jugada éstos giran de forma aleatoria durante un tiempo muy corto y al final muestra cada una de ellas uno de los símbolos. La ruleta hará 1 tirada aleatoria para cada disco con valores de la 'A' a la 'D', 4 posibilidades.

Tabla Premios -AAA - 15 monedas -BBB - 10 monedas -CCC - 5 monedas -DDD - Duplica monedas actuales
--

El jugador inicialmente recibe un bono gratis para entrar en nuestro casino de 10 monedas. El jugador tirará hasta que se acaben las monedas o hasta que acumule mas de 20 monedas allí la máquina tragaperras no te dejará jugar más.

generar_simbolo_disco (0,5)

Devuelve el valor aleatorio para un solo disco.

Entrada no recibe nada.

Devuelve el valor del disco sacado

calcular_premio() (0,5pts)

recibe 3 parámetros de entrada que son el valor de los 3 discos de la tragaperra y otro que son las monedas que tienes;

Devuelve la cantidad ganada.

Desarrollo del juego (0,75pts)

Las monedas iniciales son siempre 5.

Terminara el programa cuando no podamos jugar mas o hayamos conseguido el tope de dinero permitido por el casino (20 euros)