

Ejercicios de repaso de objetos -c2425

EjObj1- (dificultad:2) Añadir una clase Cuadrilatero a la aplicación de Figuras vista en clase. Añade como atributos los puntos que necesites.

Crea un método que te indique si el cuadrilatero tiene o no los 4 lados iguales

Crea un método que te indique si el cuadrilatero es de tipo rectángulo (tiene los lados opuestos iguales). No importa el ángulo en el que estén los lados.

EjObj2- (dificultad:5) Se desea realizar una aplicación para gestionar una competición de tenis.

Crear una clase **Jugador** que tenga como atributos, nombre y los puntos del ranking

Crear una clase **Set (de tenis)** que tiene como atributos 2 números enteros entre 0 y 7. Para que la puntuación de un set sea válida uno de los números tiene que ser 6 o 7. Si uno de los números es 6 el otro debe ser 7 o 4 o menor. Cuando se genere la puntuación de un set, de forma aleatoria, se tendrá en cuenta esta condición.

Crear una clase **Partido** que tenga como atributos 2 jugadores y la puntuación en 3 sets (se juega al mejor de 3 sets).

Al generar un partido se irán generando la puntuación de cada uno de los sets y si el mismo jugador gana los 2 primeros no será necesario generar puntuación para el tercero.

La puntuación del set se generará punto a punto, es decir, si en el set van 2-2 se generará un número aleatorio, que tendrá en cuenta el ranking del jugador (a mayor ranking más posibilidades de ganar) para saber quién gana el siguiente punto del set. Cuando la puntuación del set llegue a 6, en alguno de los jugadores, se finalizará el set si el otro tiene 4 o menos, si llegue a 6 y el otro tiene 5 se podrá jugar otro punto adicional, si se llega a un 7-5 se finaliza el set pero si se llega a un 6-6 se debe jugar un último set para finalizar con un 7-6 o 6-7

Al jugador que gane el partido se le suman 100 pts a su ranking y al que pierda se le restan 100pts.

Ejercicios de repaso de Estructuras de Control- C2425

EjCtrl1.-(Dificultad :1) Realizar un método, en la clase Fracción, que compare 2 fracciones y devuelva la mayor de ellas. El método se invocará con `fracción1.esMayorQue(fracción2)`. Invocar al método reducir fracción para utilizar la fracción reducida como base de la comparación.

EjCtrl2. (Dificultad 1) Método que reciba una cantidad de dinero en euros como parámetro y muestre por pantalla su desglose en billetes y monedas para que el número de billetes y monedas sea el menor posible. Puede hacerse sin bucles ni condicionales. Por ejemplo 37,19€ serían 1 billete de 20€, 1 billete de 10€, 1 billete de 5€, 2 monedas de 1€, 3 monedas de 0.05€ y 2 monedas de 0.02 €

EjCtrl3. (Dificultad 1) Realizar un programa que solicite números por teclado hasta que se dé un 0. Cada vez que pida un número deberá indicar el número de orden del número solicitado. Al final mostrará un mensaje con los números solicitados.

Ejemplo de ejecución si el usuario introduce los números 7 11 y 0

Dar el número 1º: 7

Dar el número 2º: 11

Dar el número 3º: 0

El usuario ha introducido 2 números antes de finalizar

EjCtrl4. (Dificultad 1) Modificar el programa anterior para que, después de que el usuario introduzca cada número se le muestre la suma de los números introducidos hasta el momento. Así después de introducir el 7 mostrará suma:7 y después de introducir el 11 mostrará suma: 18 y después de introducir el 0 mostrará 18

EjCtrl5. (Dificultad 1) Modificar el programa anterior para que, al final de la ejecución, se muestre la suma de los números pares que hay entre los introducidos por el usuario.

EjCtrl6. (Dificultad 2) Realizar un programa que muestre por pantalla la suma de los dígitos de un número (Se recomienda separar los dígitos de un número dividiendo entre 10 y tomando por una parte el cociente y por otra el resto. Los dígitos no tienen porque mostrarse en orden. Ejemplo de l número 4571 mostrará Suma:17

EjCtrl7. (Dificultad 2):Realizar un programa que muestre todos los números enteros entre 1000 y 3000 que tengan el dígito que introduzca el usuario por teclado. Se recomienda realizar un método que reciba un número y un dígito y devuelva true si el dígito está en el número.

EjCtrl8. (Dificultad 2):Realizar un programa que lea un número entero de teclado y muestre un tablero cuadrado con los n en los bordes y b en el interior. Ejemplo si recibe el número 5 escribir

```
nnnnn
nbbbn
nbbbn
nbbbn
nnnnn
```

Ejercicios de repaso de Arrays- C2425

A1. (dificultad 2) Realizar un programa que genere 10 números enteros aleatorios entre 1 y 10 y los guarde en un array. Después pedirá repetidamente y hasta que el usuario pulse un 0, un número e indicará cuantas veces aparece ese número en el array.

A2.- (dificultad 2) Realizar un programa que genere 20 números enteros aleatorios entre 1 y 10 y después indique cuál es el que aparece más veces. En caso de que haya varios que aparecen el mismo número de veces se mostrará el primero introducido.

A3.- (dificultad 2) Realizar un programa que lea 10 números enteros y después muestre los que están por encima de la media.

A4.-(dificultad 2) Realizar un programa que muestre un menú con las siguientes opciones:

- 1.- Añadir número real de tipo double (lo almacenará en un array)
- 2.- Listar números leídos. Mostrará su posición y el número
- 3.- Modificar número. Pedirá la posición y el nuevo número real y sustituirá el antiguo por el nuevo.
- 4.- Borrar número. Pedirá la posición y eliminará el número. Se considerará que ya no forma parte de los incluidos en el array a la hora de listarlos o modificarlos.
- 5.- Consultar posición. Pedirá la posición y mostrará el número que está en esa posición
- 6.- Consultar número. Pedirá un número real e indicará si está o no en el array. Si está varias veces indicará cuántas.
- 7.- Media. Mostrará la media de los números almacenados.
- 0.- Finalizará el programa

EjArrays1(dificultad 4) (Array bidimensional) Se desea almacenar los datos de ventas realizadas por los 10 empleados de una empresa en cada uno de los 12 meses del año. Para ello se va a utilizar una matriz bidimensional de números reales. En la aplicación de gestión de las ventas se dispondrá de un menú con las siguientes opciones:

- 1.- **(Dif:1)** Añadir una venta. Se pedirá la cantidad vendida, el número de vendedor (entre 0 y 9) y el mes de la venta (entre 1 y 12) y se añadirá la cantidad a la que figure en la casilla correspondiente de la matriz.
- 2.- **(Dif:2)** Mostrar la casilla (vendedor y mes) en la que hay un valor de ventas más alto. Si hay varias con la misma cantidad se mostrará la posición de una cualquier de ellas.
- 3.- **(Dif:2)** Calcular el total de ventas realizadas por la empresa (todos los vendedores en todos los meses).
- 4.- **(Dif:2)** Mostrar, dado un número de vendedor, el desglose de las ventas de ese vendedor en cada mes y la suma de ventas (total anual)
- 5.- **(Dif:2)** Mostrar, dado un mes, las ventas que ha realizado cada vendedor en ese mes.
- 6.- **(Dif:3)** Mostrar el número vendedor que tiene una cantidad en ventas en el año menor. Si hay varios con la misma cantidad se mostrará el de menor número)
- 7.- **(Dif:3)** Mostrar el mes en el que ha habido mayor cantidad vendida (suma de las ventas de todos los vendedores en el mes) . Si hay varios con la misma cantidad se mostrará el más reciente, suponiendo que estamos en diciembre.
- 8.- **(Dif:4)** Mostrar el número del vendedor que ha tenido, la mejor racha de ventas en 3 meses consecutivos.
- 0.- Fin de la aplicación

Se recomienda tener una clase Empresa que tenga un atributo con la matriz correspondiente.

EjArrays2(dificultad 4) Se desea realizar una aplicación para el control de una máquina de reparto de bebidas que tendrá en cuenta el número de monedas de 1,2, 5, 10, 20 y 50 céntimos y de 1 y 2 euros y el número de bebidas de cada tipo (refrescos de 3 tipos, agua, bebida energética, y cerveza)

Realizar una aplicación con 2 menús, uno para el usuario (que selecciona la bebida (se le mostrará el precio y si hay o no disponibles) , introduce las monedas (indicando cuántas de cada tipo), y la obtiene, junto con el cambio) y otro para el repartidor que añade bebidas a la máquina y recauda el dinero almacenado en esta.

Debes incluir, al menos, los siguientes métodos:

- Método **cantidadTotalAlmacenada**, que devuelva, como double, la cantidad de dinero total almacenada en la máquina (para el menú del repartidor)
- Método **compraBebida**, que reciba como parámetro el número de monedas introducidas de cada tipo (crear una clase **GrupoMonedas** para este dato) y el precio de la bebida en euros seleccionada y devuelva (return) el número de monedas que se deben devolver de cada tipo (objeto de tipo GrupoMonedas) restando las monedas devueltas de la cantidad total almacenada en la máquina.

EjArrays3- (dificultad:4). Se desea realizar un juego de lucha con diferentes personajes.

Crear una clase Arma que tendrá como atributos 2 enteros entre 0 y 100 que reflejan la potencia de ataque y la potencia de defensa (un ejemplo de arma sería un escudo con potencia de ataque 10 y de defensa 80)

Crear la clase Personaje que tendrá como atributos, el nombre, el porcentaje de vida, 2 posibles armas y la fuerza

En el juego se podrán, con un menú, crear personajes y armas y asociar armas a personajes, además se podrá realizar luchas entre los personajes, que se dilucidarán en función de la fuerza del personaje y de el arma (de entre las 2 posibles que tenga) que utilice en la lucha. En la lucha se utilizará aleatoriedad para dirimir el ganador de la lucha. Una vez finalizada la lucha el personaje perdedor perderá un porcentaje de su vida, pudiendo llegar a morir (0 de vida).

EjArrays4- (dificultad:4). Crear una competición con varios jugadores y partidos utilizando las clases del juego de tenis de los ejercicios de objetos.