

Ejercicios clase 2:

1 – Calculador MRUV

Sean los datos:

- Xi: Posición inicial (Km).
- Vi: Velocidad inicial (km/h).
- t: Tiempo transcurrido (s)
- a: Aceleración (m/s²)

Obtener:

- Xf: posición final (km)
- Vf: Velocidad final (km/h).

Sabiendo que:

- $xf = xi + vi \cdot t + \frac{1}{2} a (t^2)$
- $vf = vi + a \cdot t$

Nota: Para cada dato (ya sea iniciales o resultados) definir el tipo de dato a utilizar y justificar por qué se eligió.

2 – Listado de alumnos

Se desea un programa que permita ingresar los datos APELLIDO en forma de cadena de caracteres y promedio como número con decimales de una cantidad desconocida de alumnos.

Finalizado el proceso de carga, el programa deberá informar:

- La cantidad de alumnos ingresados.
- Promedio general de los alumnos.
- Apellido y promedio del alumno con mejor promedio (si hay más de uno, basta con usar el primero de la lista).

3- Librería

Se quieren registrar los artículos de una librería. Cada libro cuenta con la siguiente información:

ISBN: (International Standard Book Number) Número entero de 13 dígitos decimales.

Signatura Topográfica: Cadena de caracteres. Código alfanumérico que indica la ubicación en una biblioteca de una obra.

Costo: Número real (punto flotante de precisión simple).

Solicitar iterativamente y registrar, los datos de un conjunto de extensión desconocida de libros; finalizando dicho proceso de carga con el ingreso de un número de ISBN de valor igual a 0 (cero), para cuyo caso no deberán solicitarse ni registrarse los datos restantes Signatura Topográfica y Costo.

Informar ISBN y costo del libro más barato ingresado.

En caso de no haberse ingresado libro válido alguno, en lugar de lo indicado en el ítem anterior, informar la leyenda “No se ingresaron libros.”.

4 – Triángulos

Realice una aplicación de consola en C# que solicite la longitud (flotante de precisión simple) de cada uno de los tres lados de un triángulo. Conforme la magnitud de cada uno de ellos, empleando la menor cantidad de estructuras de control posibles, el programa deberá informar si se trata de un triángulo equilátero, isósceles o escaleno. Tenga presente que es menester verificar que el valor ingresado para cada uno de los lados sea positivo.

2. Para avanzar, es importante saber que no cualquier terna de valores constituye un triángulo. Al ejercicio anterior, agréguele la verificación de que los tres valores ingresados corresponden a un triángulo, de modo que sólo informe tipo si efectivamente se trata de un triángulo. Pista: sean a , b y c los lados de un triángulo: Si $a+b>c$ y $a+c>b$ y $b+c>a$, entonces abc constituye un triángulo.

3. Modifique el programa anterior para que permita operar analizando una cantidad desconocida de ternas de valores (se sabe que dicha cantidad no excede las 65535 ternas). El programa deberá solicitar iterativamente lado a , lado b y lado c hasta que se ingrese una terna que no corresponda a un triángulo. Cuando ello suceda, el programa deberá informar:

- a. El total de triángulos válidos ingresados.
- b. La cantidad de triángulos de cada tipo.
- c. El porcentaje de triángulos de cada tipo.

Realizar (por ahora) solo en consola ambos ejercicios.

5 – Equipo de futbol

Realice una aplicación de consola de C# que solicite la posición (arquero, defensor, mediocampista o delantero), cantidad de goles (para los arqueros se pedirá la cantidad de atajadas), cantidad de partidos, nombre y apellido y DNI de una cantidad indeterminada de jugadores de futbol.

Una vez ingresados, mostrar los mejores jugadores por promedio de goles y de atajadas.