



Question - 1  
Tienda de artículos de playa

SCORE: 25 points

Cuando Lily Snaderstein visitó Costa Rica, se enamoró de sus costas y se quedó a vivir aquí. Abrió una tienda de artículos de playa en Carrillo, Guanacaste. Hace un par de años inauguró una sucursal en Liberia, y recientemente otra en Sardinal. Al inicio, saber la buena salud de su negocio era fácil para Lily pues era sólo una sucursal. Pero ahora se ha complicado de varias formas.

Lily ha sido muy meticulosa en llevar los registros de ventas en libros contables, un libro por cada tienda. En los libros, sus empleados anotan cada producto vendido, y al final del día, el total de ventas tras realizar el cierre de caja. Desdichadamente los libros contables están en papel. Lily ha pagado un proceso de digitación y actualmente se tienen los cierres de caja de cada una de las tres tiendas. El siguiente es un trozo del registro digitado:

Ejemplo de entrada:

```
28/06/2017 Carrillo 12490
13/07/2017 Sardinal 250995
14/07/2017 Sardinal 67913.5
14/07/2017 Liberia 196455
17/08/2017 Carrillo 29000
```

Cada línea del registro contiene un cierre de caja de una sucursal para un día particular. Por ejemplo, la primera línea indica que el 28 de junio de 2017, la sucursal de Carrillo vendió en total 12490 colones. Debe tenerse en cuenta que la digitación se realizó en función de la disponibilidad de los libros contables, pues los del último año están en uso aún. Por tanto, los cierres de caja pueden venir en cualquier orden en la entrada.

La estadística inmediata que más le podría a ayudar a Lily para tener un panorama general de la salud de sus negocios y poderlos comparar, es el promedio de ventas de cada sucursal. Sin embargo, dado que su negocio es dependiente del sector turismo, las estadísticas se afectan fuertemente por la temporada alta y baja de turismo. Lily desearía que los promedios se reporten en una tabla por cada sucursal tanto en temporada baja como alta. La *temporada alta* corresponde a los meses de vacaciones escolares hasta semana santa (diciembre a abril, inclusive) y las vacaciones de medio año (julio). La *temporada baja* corresponden a los demás meses.

Ejemplo de salida:

TIENDA	BAJA	ALTA
=====	=====	=====
Carrillo	20,745.00	0.00
Liberia	0.00	196,455.00
Sardinal	0.00	159,454.25

La tabla debe contener en las filas las tres sucursales ordenadas alfabéticamente. Los nombres de las sucursales se deben alinear a la izquierda. Los promedios de ventas en temporada baja y alta se deben imprimir en columnas de ancho 15, alineadas a la derecha. Los valores monetarios se deben imprimir con separadores de miles y dos decimales para facilitar a Lily la lectura.

## Question - 2

### Frecuencias en sorteos de azar

SCORE: 50 points

Classes Object Array

Probablemente le haya pasado que si quiere "jugarse un pedacito" de lotería para el "gordo navideño", algún revendedor le haya subido el precio si el número "es bajo". La gente piensa que "los números bajos" salen con más frecuencia y los revendedores se aprovechan de ello porque saben que los compradores pagarán sus caprichos. ¿Estará la gente en lo cierto con esta creencia?

Escriba un programa que ayude a comprobar o rechazar la hipótesis de que los números bajos salen con más frecuencia que los altos. Para efectos de este análisis se considerará un **número bajo** aquel entre 00 y 49, y un **número alto** aquel entre 50 y 99. Los sorteos a analizar constarán de tres premios. Por ejemplo, en la entrada siguiente se analizarán **tres** tipos de sorteos de la Junta de Protección Social (JPS) de Costa Rica: lotería nacional de los domingos, chances de los martes, y chances de los viernes. Esta cantidad de tipos de sorteos es el primer número en el archivo de entrada.

#### Ejemplo de entrada:

3	NOMBRE	DS	DD	MM	AAAA	NUM1	SERIE1	NUM2
SERIE2	NUM3	SERIE3						
2	Chances	V	22	6	2018	68	176	50
868	80	268						
1	Chances	K	19	6	2018	02	174	57
257	17	731						
3	Loteria	D	17	6	2018	90	096	94
795	26	183						
2	Chances	V	15	6	2018	96	421	11
035	66	428						
1	Chances	K	12	6	2018	18	863	93
635	77	601						
3	Loteria	D	10	6	2018	48	115	29
071	27	687						
2	Chances	V	08	6	2018	38	986	60
694	55	39						
1	Chances	K	05	6	2018	89	291	45
354	26	428						
3	Loteria	D	03	6	2018	80	307	84
268	99	577						

La primera línea de entrada corresponde a un encabezado de un archivo de valores separados por espacios (CSV). El primer número en esa línea indica la cantidad de tipos distintos de sorteos (para la entrada de ejemplo es 3: chances de los viernes, chances de los martes, lotería de los domingos). El resto de columnas se detallan en la siguiente tabla.

Columna	Detalles
#	Cantidad de tipos de sorteos
NOMBRE	Nombre del sorteo (ej: Loteria, Chances)
DS	Día de la semana (ej: D=domingo, K=martes, V=viernes)
DD	Día del mes (01-31)
MM	Mes (01-12)
AAAA	Año
NUM1	Número ganador del primer premio
SERIE1	Serie del número ganador del primer premio
NUM2	Número ganador del segundo premio
SERIE2	Serie del número ganador del segundo premio
NUM3	Número ganador del tercer premio
SERIE3	Serie del número ganador del tercer premio

El analizador a implementar debe reportar frecuencias de aparición de los números bajos y altos de todos los tipos de sorteos recibidos en la entrada estándar. Las frecuencias se reportan por separado para el primer premio (BAJO1 y ALTO1), segundo premio (BAJO2 y ALTO2), y tercer premio (BAJO3 y ALTO3).

#### Salida de ejemplo:

NOMBRE	DS	BAJO1	ALTO1	BAJO2	ALTO2	BAJO3	ALTO3
Chances	K	67%	33%	33%	67%	67%	33%
Chances	V	33%	67%	33%	67%	0%	100%
Loteria	D	33%	67%	33%	67%	67%	33%

Su solución debe ser completamente orientada a objetos. Para implementarla siga estos lineamientos.

1. Su solución debe constar de al menos tres clases. Se sugieren: `Solution`, `AnalizadorLoteria` y `TipoSorteo`.
2. La clase `Solution` ya está provista. Estúdiela para comprenderla. Su trabajo es simplemente instanciar un objeto de la clase controladora `AnalizadorLoteria` e iniciar el proceso de análisis de datos.
3. La clase `AnalizadorLoteria` está parcialmente implementada. La clase tiene como atributos los archivos donde se recibirán los datos de entrada y donde se deben imprimir las estadísticas de salida. Los métodos `correr()`, `abrirArchivoEntrada()` y `abrirArchivoSalida()` se encargan de abrir y cerrar estos archivos. Estúdielos para comprenderlos.
4. Inicie la implementación de su solución en el método `analizarDatos()` de la clase `AnalizadorLoteria`. Deberá leer la cantidad de tipos de sorteos del archivo de entrada. Dado que esta cantidad es variable de acuerdo a la entrada, deberá crear un arreglo de `TipoSorteo`. Declare el arreglo como atributo de la clase.

5. Inicialice el arreglo de tipo de sorteos. Dado que `TipoSorteo` no es un tipo de datos primitivo, Java crea un arreglo de referencias nulas. Debe recorrer el arreglo creando un objeto `TipoSorteo` en la memoria dinámica y cambiar cada referencia en el arreglo para que apunte a su respectivo objeto.
6. Lea el resto del encabezado desde el archivo de entrada. Dado que todos los archivos en la entrada tienen las mismas columnas, puede ignorar las etiquetas en los encabezados. Note que leer e imprimir el encabezado es responsabilidad de la clase `AnalizadorLoteria`.
7. Mientras hayan registros (líneas) en el archivo de entrada, lea el número del tipo de registro. Utilice este número para localizar el objeto `TipoSorteo` correspondiente en el arreglo (recuerde que los índices de los arreglos en Java inician en 0). Asigne la responsabilidad al `TipoSorteo` de leer el resto de datos de la línea.
8. Un objeto de la clase `TipoSorteo` representa uno de los tipos de sorteos que se quieren analizar, por ejemplo los chances del martes en el ejemplo de entrada. Este objeto debe saber identificarse, algo como `"Chances"` y `'V'` para que pueda distinguirse de otros cuando se imprima la tabla de estadísticas.
9. Un objeto `TipoSorteo` debe además ser responsable de llevar el conteo de números bajos y de números altos que resultaron ganadores de ese tipo de sorteo. Cuando el objeto lee un sorteo desde una línea de entrada, actualizará sus contadores para poder calcular las frecuencias de aparición de los números bajos y altos de cada uno de los tres premios.
10. Una vez leídos todos los sorteos (líneas) de la entrada, el objeto controlador `AnalizadorLoteria` deberá imprimir las estadísticas siguiendo el formato del ejemplo de salida. El `AnalizadorLoteria` es el responsable de imprimir el encabezado, pero debe delegar la responsabilidad de imprimir las frecuencias a cada tipo de sorteo del arreglo.
11. Cuando se pide a un `TipoSorteo` imprimir las frecuencias, este debe calcular los porcentajes con aritmética flotante e imprimirlos redondeados a cero decimales. Si un `TipoSorteo` nunca tuvo sorteos (datos en la entrada), no debe imprimirse en las estadísticas.

Para el ejemplo de salida puede verse que los números bajos no salen con más frecuencia que los altos, lo que rechaza la hipótesis, al menos con este pequeño tamaño de muestra.

### Question - 3

#### Vectores en el plano cartesiano

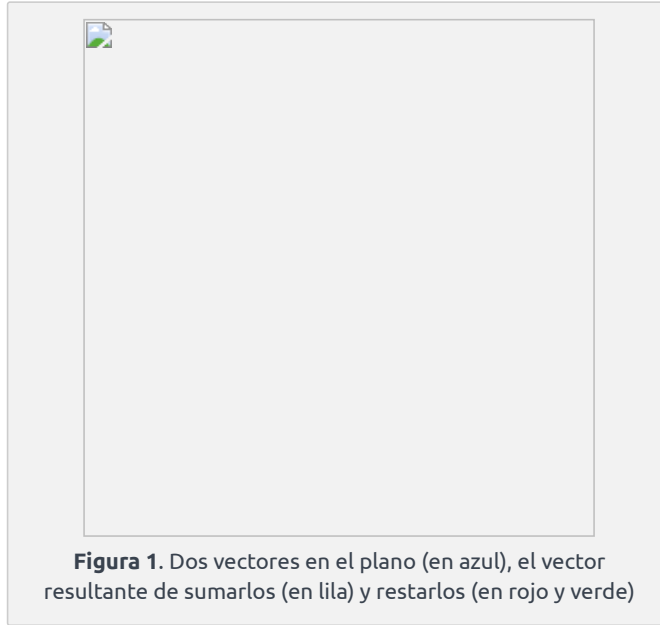
SCORE: 48 points

Class Instance Operator Overloading

Sus amigos de secundaria quieren que les ayude con un poco de física. Ellos tienen parcialmente implementado, con algunos errores, un programa para realizar operaciones con vectores en el plano cartesiano. Por favor, ayude a corregirlo y completarlo.

Implemente una clase `Vector` para representar cantidades vectoriales físicas (como la velocidad, la fuerza y el campo eléctrico) en el plano cartesiano. Un vector consta de magnitud y dirección, y hay varias convenciones para representarlas. Su clase `Vector` debe representar

los vectores en el plano cartesiano, con origen siempre en el punto  $(0, 0)$  y como punto final las coordenadas  $(x, y)$ , como se aprecia en la Figura 1.



Su clase `Vector` debe ser capaz de realizar lo siguiente

1. Almacenar los componentes  $(x, y)$ , los cuales pueden ser distintos de un vector a otro. Inicializarlos por defecto como el vector nulo  $(0, 0)$ .
2. Leer un vector de la entrada estándar como una pareja de valores separados por un espacio que representan coordenadas cartesianas. Su método de lectura puede retornar `true` cuando se ha leído un vector distinto al vector nulo.
3. Calcular la norma  $r$  o magnitud  $|\vec{V}|$  utilizando el Teorema de Pitágoras.
4. Calcular la dirección como una coordenada angular  $\theta$  en radianes. Sugerencia utilice el método `Math.atan2(y, x)` que retorna  $\theta$ .
5. Imprimir el vector en la salida estándar como coordenadas cartesianas  $(x, y)$ .
6. Imprimir el vector en la salida estándar como coordenadas polares  $(r, \theta)$ .
7. Sumar y restar vectores. La suma o la resta de dos vectores  $\vec{V}_1 = (x_1, y_1)$  y  $\vec{V}_2 = (x_2, y_2)$  es el vector resultante de sumar o restar sus componentes:

$$\vec{V}_1 + \vec{V}_2 = (x_1 + x_2, y_1 + y_2)$$

$$\vec{V}_1 - \vec{V}_2 = (x_1 - x_2, y_1 - y_2)$$

Sus amigos de secundaria quieren indicar en la primera línea, si se quiere imprimir en coordenadas cartesianas (`cartesian`) o polares (`polar`). Luego el programa lee dos vectores en el plano. Por sencillez, los vectores siempre se leen en coordenadas cartesianas. Luego de leer los vectores, se debe mostrar su magnitud y dirección de acuerdo al sistema de coordenadas indicado en la primera línea.

Finalmente el programa calcula e imprime la resultante de sumarlos y restar ambos vectores como se muestra en el ejemplo de ejecución. Dado que sus amigos quieren usar el programa para revisar ejercicios de libro, haga que siempre que el programa imprima un número entero, se haga sin decimales; y si no es entero, con dos decimales.

#### Ejemplo de entrada:

```
cartesian  
  
2 5  
4 -3
```

#### Ejemplo de salida:

```
V1 = (2, 5)  
V2 = (4, -3)  
  
V1 + V2 = (6, 2)  
V1 - V2 = (-2, 8)  
V2 - V1 = (2, -8)
```

### Question - 4

#### Calculadora fraccional estándar

SCORE: 50 points

Operator Overloading

Class

Simule una calculadora fraccional sin prioridad de operaciones. La calculadora trabaja en lote y no interactivamente. Es decir, la calculadora recibe expresiones aritméticas con fracciones en la entrada estándar, las replica en la salida estándar e imprime el resultado de la operación. Una expresión es una combinación de operadores y fracciones que termina en el signo de igualdad. La calculadora debe leer operaciones hasta encontrar el fin de archivo.

#### Entrada de ejemplo:

```
5/-4 - 2/3 =  
-1/1 + 4/9 * 3/-1 + 7/13 =
```

#### Salida de ejemplo:

```
-5/4 - 2/3 = -23/12  
-1 + 4/9 * -3 + 7/13 = 86/39
```