

# Examen 1

*Duración: 3 horas. IDE: IntelliJ Community Edition.*

## 1. [10 puntos] Preparativos para el examen.

1. Cree un repositorio privado en GitLab UCR o Git con el siguiente nombre **ci-0112-2022a-examen1-carnet**. Agregue al profesor como Maintainer o colaborador. Clone el repositorio en su máquina local. Recuerde configurar el usuario del repositorio local (nombre y correo).
2. Cree un proyecto de Java (Gradle) utilizando IntelliJ (Community Edition) en la carpeta que acaba de clonar.
3. Cree los directorios que le hacen falta al proyecto (*design, doc, tests*).
4. Cree un *package* utilizando el URL de la UCR (como se ha hecho en el curso).
5. Dentro del *package* cree una clase llamada *Solution*. En ella cree el método *main* del proyecto. Pruebe que su proyecto funciona imprimiendo en la entrada estándar el siguiente mensaje: "Hola mundo". Luego, comente esa línea donde se realiza la impresión.
6. Cree un método *run*. Usted puede utilizar este método para probar los métodos que tendrá que implementar más adelante. Durante la evaluación de su examen el profesor utilizará este método para ejecutar sus métodos.
7. Cree un archivo *.gitignore* para que se ignoren al subir el proyecto al repositorio de control de versiones las carpetas que se generan automáticamente (*.gradle, .idea, build, doc*).
8. Realice el primer *commit* de su proyecto. Luego, suba su servidor (GitLab UCR o GitHub) el proyecto.
9. Debe enviar un correo al profesor con el enlace a su repositorio.

2. [20 puntos] Implemente un método que lea de la entrada estándar una o varias filas de números separados por espacios. El primer número de cada fila indica la cantidad de números en la fila (sin contar el primer número). Su subrutina debe de imprimir en la salida estándar la cantidad mayor de números enteros consecutivos ascendentes (de forma ininterrumpida) de cada una de las filas. Un ejemplo de cuatro números enteros consecutivos ascendentes es [33, 34, 35, 36]. Un ejemplo de una fila de cuatro números enteros **no consecutivos** es [33, 34, 36, 37]. En el ejemplo anterior hizo falta el 35. La tabla 1 muestra un ejemplo de entrada.

*Tabla 1. Ejemplo de una entrada.*

10	1	2	3	4	5	6	7	8	9	20										
20	1	2	34	4	5	6	7	8	9	20	21	22	23	24	25	26	7	8	9	20

La tabla 2 muestra la salida esperada al procesar las filas de números de la tabla 1.

*Tabla 2. Salida esperada al procesar los datos de la tabla 1. La salida mostrada en esta asume que la entrada fue tomada de un archivo de texto redirigido a la entrada estándar.*

9
7

3. [20 puntos] Implemente un método que lea de la entrada estándar uno o más pares de filas ordenas de la siguiente manera. La primera fila contiene un texto; la segunda fila contiene un subtexto. Cada par de filas está separado por una fila/línea en blanco. Su método debe de contar cuantas veces aparece el subtexto (de la segunda fila) en el texto (de la primera fila) e imprimirlas como se muestra en la tabla 4.

*Tabla 3. Ejemplo de una entrada.*

```
Pepito compra chocolates. Pepito invita a sus amigos a comer chocolate.
Pepito

Me gusta comer cas. ¡Quiero comer cas, cas, cas, cas, cas!
cas
```

La tabla 4 muestra la salida esperada al procesar los pares de filas de la tabla 3.

*Tabla 4. Salida esperada al procesar los datos de la tabla 3. La salida mostrada en esta tabla asume que la entrada fue tomada de un archivo de texto redirigido a la entrada estándar.*

Palabra	Total
-----	-----
Pepito	2
cas	6

Para resolver este ejercicio debe utilizar ciclos y estructuras condiciones (*if/else*). Está prohibido utilizar métodos como *contains* (o similares) de la clase *String* para encontrar la cantidad de veces que se repite la palabra.

4. **[50 puntos]** Implemente un método que le ayude al dueño de una pequeña pulpería a procesar sus facturas. Las facturas poseen la lista de productos que se vendieron (el nombre y el precio del producto). Además, la pulpería posee un sistema de cupones que otorgan un descuento de un cierto porcentaje a ciertos productos.

Debido a la falta de un sistema de facturación, el pulpero escribió el borrador de cada factura a mano y escribió cada factura en una línea. El formato que se siguió para escribir los borradores de las facturas fue el siguiente.

- Una letra C al inicio de la línea indica que se utilizó un cupón. Seguido de la letra C se indica el porcentaje del descuento. Una letra S al inicio indica que la compra se realizó sin un cupón. En este caso no se incluye el porcentaje del descuento.
- Luego, se listan los productos comprados. Por cada producto se muestra su nombre (compuesto de una sola palabra) y el precio del producto. Cuando se utiliza un cupón, los artículos que poseen descuento se marcan con un menos (-) adelante del precio.

La tabla 5 muestra un ejemplo de entrada esperada.

*Tabla 5. Ejemplo de entrada esperada.*

```
C 50.0 Atun 100.0 Frijoles 50.0 Helados -100.0 Leche -100.0
S Atun 100.0 Frijoles 50.0 Helados 100.0 Leche 100.0 Huevos 75.0 Pan 25.0
```

Su programa debe imprimir en la salida estándar las facturas de forma ordenada y legible. Su programa debe tomar la información provista por el pulpero y darle el siguiente formato:

- Un encabezado con el número de factura, empezando en 1.
- En la segunda línea, se indica si se utilizó un cupón y el porcentaje del descuento
- Tercera línea, un separador.
- Luego, listar los nombres de los artículos, los precios sin aplicar el descuento y los precios con el descuento; una línea por cada producto.
- Luego, después del último producto, imprimir un separador.
- Luego, imprimir el subtotal que incluye el total sin el descuento y el total del monto a descontar.
- Luego, el total final aplicando el descuento.

- Por último, imprimir un separador entre cada factura.

Nota. Recuerde que el ciclo *while* se puede utilizar cuando se desconoce el número de iteraciones se deben ejecutar y estos se pueden anidar.

La tabla 6 muestra la salida esperada al procesar los datos de la tabla 5.

*Tabla 6. Salida esperada al procesar los datos de la tabla 5. La salida mostrada en esta asume que la entrada fue tomada de un archivo de texto redirigido a la entrada estándar.*

```
FACTURA 001
Con cupon 50%
ARTICULO    PRECIO  DESCUENTO
-----
    Atun      100.00
    Frijoles   50.00
    Helados    100.00    50.00
    Leche      100.00    50.00
-----
    Subtotal   150.00   -100.00
    Total      250.00
=====
FACTURA 002
Sin cupon
ARTICULO    PRECIO  DESCUENTO
-----
    Atun      100.00
    Frijoles   50.00
    Helados    100.00
    Leche      100.00
    Huevos     75.00
    Pan        25.00
-----
    Subtotal   450.00    -0.00
    Total      450.00
=====
```

Notas importantes:

- El profesor le va a solicitar cada cierto tiempo que realicé un *commit* al sistema de control de versiones que esté utilizando.
- Se recomienda utilizar los casos de prueba provistos por el profesor para verificar el correcto funcionamiento de sus métodos.