Universidad de Costa Rica
Escuela de Ciencias de la Computación e Informática
CI-0202 Principios de Informática
Ampliación, Grupo 12
II Semestre 2020, 11 de diciembre 2020
Docente Sivana Hamer

Declaro bajo juramento que realizaré esta evaluación de forma **estrictamente individual**. En caso de que se sospeche lo contrario, estoy consciente de que se me aplicará el debido proceso estipulado en el Reglamento de orden y disciplina de los/las estudiantes de la Universidad de Costa Rica, el cual plantea sanciones de suspensión hasta por seis años.

Instrucciones: El examen es de elabroación **completamente individual** realizado el viernes 11 de diciembre del 2020 de 7:00 am a 9:00 am de manera sincrónica. Resolverá el problema planteado en el enunciado como si lo estuviera resolviendo en un cuaderno de examen físico. Escriba el código en Python 3 que soluciona el problema planteado con bolígrafo en su cuaderno o una hoja de papel. Si necesita hacer correcciones, puede tachar, siempre que el resultado sea claro. Al finalizar el examen, debe tomar fotografías legibles de todo el examen, crear un archivo PDF con todas las imágenes ordenadas, y subir el archivo pdf el viernes 11 de diciembre del 2020 antes de las 9:00 am al aula virtual. A menos que esté escrito explícitamente, no será necesario realizar validación de datos de entrada. Suponga que los datos que se ingresan siempre son válidos. En cada punto, se evaluará la correctitud, completitud, y buenas prácticas de programación. Estas incluyen, pero no se limitan a identificadores significativos para variables e indentación adecuada. Además, si considera que requiere de más funciones para solucionar el examen, puede implementarlas.

Enunciado

La compañía Dotify provee servicios de transmisión de audio. Específicamente su servicio más popular es que tienen disponible millones de canciones de distintos artistas que cualquier usuario/usuaria del servicio puede reproducir. Dado la gran cantidad de artistas que suben sus canciones al servicio, están interesados en mostrar información sobre la popularidad de los artistas que se encuentran disponibles en sus servicios.

La información sobre los/las artistas de la plataforma se guarda en un archivo nombrado `canciones-artistas.csv`. Cada línea del archivo cotiene la información sobre una canción de un/una artista. Una línea incluye el nombre de el/la artista, el nombre de la canción y la cantidad de veces que se ha reproducido una canción. El siguiente ejemplo es como se vería el archivo:

```
artista;cancion;reproducciones
Ariana Grande;Dangerous Woman;701
Ariana Grande;Greedy;42
Michael Jackson;Billie Jean;621
Black Pink;WHISTLE;2001
Selena Gomez;Lose You To Love Me;433
Selena Gomez;Birthday;126
Ariana Grande;thank u, next;65
Michael Jackson;Beat It;62
Ariana Grande;Positions;19
Selena Gomez;Wolves;982
```

El programa debe listar el detalle de canciones de cada artista. Este detalle será parte de la información de cada artista y se implementará por medio de una matriz. Cada fia de la matriz representa una canción y siempre tendrá 2 columnas: el nombre y la cantidad de reproducciones. Esta es la misma información que se mostrará. Además, se calculará y mostrará para cada artista el total de canciones y el total de reproducciones a todas sus canciones. Se mostrarán los artistas en el orden de aparición del archivo. El siguiente es un ejemplo de cómo debería ser la salida del programa:

Artista: Ariana Grande
Total reproducciones: 827 Total canciones:4

Canción	Reproducciones
Dangerous Woman	701
Greedy	42
thank u, next	65
Positions	19

Artista: Michael Jackson
Total reproducciones: 683 Total canciones:2

Canción	Reproducciones
Billie Jean	621
Beat It	62

Artista: Black Pink
Total reproducciones: 2001 Total canciones:1

Canción	Reproducciones
WHISTLE	2001

Artista: Selena Gomez
Total reproducciones: 1541 Total canciones:3

Canción	Reproducciones
Lose You To Love Me	433
Birthday	126
Wolves	982

Su solución debe estar diseñada siguiendo el paradigma orientado a objetos. Se utilizará un diseño de 2 clases: `Artista` e `Dotify`. `Artista` estará encargada acerca de la información de el/la artista con sus canciones. `Dotify` estará encargada de almacenar las instancias de `Artista`, leer información del archivo, y mostrar los resultados.

Evaluación

1. (45 puntos) Implementa la clase `Artista`.
   a) (10 puntos) `Artista` tiene atributos que le permiten almacenar al artista con sus canciones.
   b) (35 puntos) `Artista` implementa los siguientes métodos:
      1) (5 puntos) Un constructor para inicializar los atributos de la clase.
      2) (5 puntos) `agregar_cancion(c, r)` que agrega `r` reproducciones a la canción `c`.
      3) (5 puntos) `calcular_total_canciones()` que calcula y retorna el total de canciones que tiene un artista.
      4) (5 puntos) `calcular_total_reproducciones()` que calcula y retorna el total de reproducciones a todas las canciones del artista.
      5) (15 puntos) `imprimir_canciones()` que muestra la información de el/la artista.
      Ejemplo:

```
Artista: Selena Gomez
Total reproducciones: 1541  Total canciones:3
-----------------------------------------
             Canción       Reproducciones
-----------------------------------------
 Lose You To Love Me                  433
            Birthday                  126
              Wolves                  982
```

2. (55 puntos) Implementa la clase `Dotify`.
   a) (5 puntos) `Dotify` tiene un atributo que le permite guardar los/las distintos artistas.
   b) (50 puntos) `Dotify` implementa los siguientes métodos:
      1) (5 puntos) Un constructor para inicializar los atributos de la clase.
      2) (15 puntos) `buscar_artista(n)` que busca la instancia de el/la artista con el nombre `n` y retorna la instancia. En el caso de que no existe, retorna una instancia nueva de `Artista`.
      3) (25 puntos) `leer_archivo()` que abre el archivo `canciones-usuarios-2020.csv`, lee la información de este, y la agrega a cada una de las instancias de `Artista`.
      4) (5 puntos) `mostrar_canciones()` que muestra la información de cada `Artista`.