

Taller 4

Montículos y Tablas de símbolos

Para el desarrollo de este ejercicio se trabajará con el dataset de repositorios de github (ver [fuente original](#)). Para el desarrollo del ejercicio se busca implementar un ADT Repositorio que describa los datos de un repositorio y realizar varios ejercicios de análisis utilizando montículos y tablas de símbolos.

El dataset se compone de un archivo separado por comas (CSV - comma separated values), donde cada línea representa los datos de un repositorio. Se debe así mismo leer los datos del repositorio y crear las instancias de Repositorio en una colección en memoria para realizar las distintas consultas que se proponen.

Ejercicios a desarrollar

1. Lectura de datos y creación de la colección de objetos Repositorio.

Implementar la función de biblioteca leerDataset que toma como entrada la ruta del archivo de datos y que retorna la colección de objetos repositorio:

```
ArrayList<Repositorio> leerDataset(String ruta)
```

Nota: Importante garantizar el tipo de datos de las distintas columnas para los ejercicios posteriores, e.g. created es Date; id, forks, stars, open_issues, subscribers son int y los demás son String.

2. Consulta de datos por usuario

Implementar la función de biblioteca datosPorUsuario que totalice sobre todos los repositorios de un usuario los siguientes datos:

- Número total de repositorios
- Total de estrellas de los repositorios del usuario
- Total de suscriptores del usuario
- Número promedio de open_issues por usuario

Para cada usuario se debe imprimir el nombre del usuario y la información correspondiente a ese usuario. La siguiente es la firma de la función:

```
void consultaPorUsuario(ST<String, Bag<Repositorio>> repoXusuario)
```

El argumento de entrada es una tabla de símbolos con llave el usuario y valor un Bag con los repositorios de ese usuario.

3. Rankings de proyectos por lenguaje de programación

Implementar una función para rankear los repositorios para cada lenguaje de programación en función del número de estrellas de cada repositorio en el correspondiente lenguaje.

Como entrada se deben pasar los datos en una tabla de símbolos con llave el lenguaje y valor la colección de repositorios de ese lenguaje. Como salida se debe entregar un árbol de búsqueda binario, con llave el lenguaje y valor una cola de mínima prioridad con los $M=10$ proyectos de mayor número de estrellas de ese lenguaje.

```
BST<String, MinPQ<Repositorio>> rankingsPorLenguaje(ST<String,
                                     Bag<Repositorio>> repoXlenguaje)
```

El cliente luego de hacer la consulta, imprime en pantalla los rankings para cada uno de los lenguajes, ordenados de mayor a menor.

Notas:

- Cada numeral vale 1 punto.
- El programa principal se debe llamar Taller4 y desde su main se llaman las funciones para leer los datos y realizar las consultas solicitadas.
- El ADT Repositorio y las funciones de biblioteca se pueden implementar en la misma clase.

Entregables:

Remitir la implementación de proyecto a través del aula digital. No incluir la biblioteca `algs4.jar`.

Nombrar el archivo comprimir `ApellidoNombre1-ApellidoNombre2-ApellidoNombre3-Taller4.zip` (o `.rar` o `.7z` o `.tgz`)

Se puede desarrollar el ejercicio en equipos de máximo 3 personas.