

Conjuntos ordenados:

- Declara y construye un conjunto ordenado (TreeSet) que almacene cadenas (String).
- Introduce varias cadenas en el conjunto.
- Realiza un recorrido (for each) del conjunto y comprueba que el iterador retorna los elementos ordenados según el orden lexicográfico de las cadenas (orden alfabético). Este comportamiento se debe a que las cadenas implementan la interfaz Comparable, es decir, tienen un orden natural.
- Implementa el orden natural de la clase Usuario (nombre, nombre real y contraseña). El orden natural debe basarse en el orden alfabético del nombre de usuario. Utiliza el método compareTo de la clase String para comparar los nombres de los usuarios.
- Declara y construye un conjunto ordenado de usuarios.
- Construye varios usuarios y añádelos al conjunto. Observa que si introducimos varios usuarios con el mismo nombre sólo el primero que se inserte permanece en el conjunto. Un conjunto ordenado determina que un elemento es repetido si en la comparación obtiene el valor 0. Por tanto, es recomendable que la implementación del orden natural sea consistente con la igualdad (equals).
- Recorre el conjunto y muestra los usuarios. Observa que el iterador retorna los elementos según el orden natural.
- Implementa un comparador de usuarios (interfaz java.util.Comparator) que ordene los usuarios según el orden alfabético de sus nombres reales.
- Declara y construye un conjunto ordenado de usuarios que aplique el orden implementado por el comparador anterior. Añade usuarios a la colección (método addAll) y realiza un recorrido para comprobar que el orden es el correcto.
- Utiliza los métodos sort de la clase Collections para ordenar la lista, según el orden natural y con el comparador de orden inverso. Muestra el contenido de la lista tras ambas ordenaciones.

1. Implementa un método que tenga como parámetro un argumento de tamaño variable de tipo String y que retorne un mapa ordenado que asocie cadenas con el número de repeticiones en el parámetro del método. Por ejemplo, si es llamado con metodo("hola", "examen", "hola") retornará un mapa ordenado que asociaría a la cadena "examen" el valor 1 y a la cadena "hola" el valor 2.

2. Escribe un método en Java que tenga como parámetro un mapa que asocie cadenas con enteros y que devuelva un conjunto con las cadenas que tienen asociado un número par.

3. La moda es el valor (o valores) de una serie de números que más veces se repite. Por ejemplo, la moda de la serie {2, 3, 3, 3, 4, 5, 4, 6, 4} es {3, 4}. En el caso de que no se repita ningún valor la serie no tiene moda. Implementa un método que reciba como parámetro una serie de números de tamaño variable y devuelva una lista que contenga la moda de dicha serie.

Una academia nos pide hacer un programa para hacer un pequeño test a sus alumnos.

Estas preguntas, para facilitar la inclusión, estarán escritas en un txt (incluido en la descarga del proyecto).

Una opción se compone de:

El texto de la opción (digamos la respuesta)

Es correcto o no

Una pregunta consta de:

Pregunta (tendrá delante dos puntos y coma ;P;)

Opciones de la pregunta (entre 2 y 4)

Opción correcta (tendrá delante dos puntos y coma ;R;)

Puntos

La pregunta no será válida en los siguientes casos:

Las opciones no están entre 2 y 4.

La opción correcta esta entre el número de opciones y es un número.

Los puntos es un número entero.

Sus métodos son:

mostrarPregunta(): muestra la pregunta con sus opciones.

comprobarRespuesta(int respuestaUsuario): comprueba la respuesta del usuario si es correcta o no.

Getter de los atributos.

Un test está formado por un conjunto preguntas y los puntos acumulados. Piensa que debemos saber por cual pregunta vamos.

Sus métodos son:

cargarPreguntas(String fichero): carga todas las preguntas del fichero

siguientePregunta(): devuelve la siguiente pregunta

reiniciarTest(): nos permite reiniciar el test.

realizarTest(): empieza el test y empieza a formular las preguntas

El fichero de preguntas tiene el siguiente formato:

;P;Pregunta 1

Opción 1 pregunta 1

Opción 2 pregunta 1

Opción 3 pregunta 1

Opción 4 pregunta 1

;R;Numero opción correcta

Puntos pregunta 1

;P;Pregunta 2

Opción 1 pregunta 2

Opción 2 pregunta 2

...

## Repaso de Segunda Evaluación

Rellenar cada ejercicio con el código correspondiente:

### Torneo de Tenis

Primero, debe pedir al usuario que ingrese los nombres de ocho tenistas.  
A continuación, debe pedir los resultados de los partidos juntando los jugadores de dos en dos. El ganador de cada partido avanza a la ronda siguiente.

El programa debe continuar preguntando ganadores de partidos hasta que quede un único jugador, que es el campeón del torneo.

El programa en ejecución debe verse así:

Jugador 1: Nadal  
Jugador 2: Melzer  
Jugador 3: Murray  
Jugador 4: Soderling  
Jugador 5: Djokovic  
Jugador 6: Berdych  
Jugador 7: Federer  
Jugador 8: Ferrer

#### Ronda 1

a.Nadal - b.Melzer: a  
a.Murray - b.Soderling: b  
a.Djokovic - b.Berdych: a  
a.Federer - b.Ferrer: a

#### Ronda 2

a.Nadal - b.Soderling: a  
a.Djokovic - b.Federer: a

#### Ronda 3

a.Nadal - b.Djokovic: b

Campeon: Djokovic

¿Y si el torneo tiene 16 jugadores? ¿Y si tiene 8 jugadores?

¿Y si son impares los jugadores? Si un jugador no tiene contrincante pasa automáticamente a la siguiente ronda.

Fechas de Nacimiento

-----

Dado un fichero con fechas de nacimiento de personas en el formato:

Ana:21/06/1990

Bea:12/02/2018  
Clara:01/01/2005

Ejercicio 1: escriba un método que diga si dos personas del fichero nacen en el mismo día (aunque sea en años diferentes)

Ejercicio 2: escriba un método que indique quién es la persona(s) más vieja según las fechas de nacimiento del fichero junto con su edad en años.

Ejercicio 3: escriba un método que indique quién es la persona(s) que tiene el primer cumpleaños del año.

### Contraseñas

Actualmente un método muy utilizado para escoger una contraseña es cambiar ciertas letras de una determinada palabra por números, por ejemplo:

me gusta el futbol.

m3 gust4 3l futb0l!

Por lo tanto, para poder hacer más fácil esta tarea, realice una función que utilizando un fichero de la forma:

e  
3  
a  
4  
o  
0

Pida una frase y la muestre con las sustituciones realizadas.

Además, como siempre se desea una contraseña mucho más segura modifique la función anterior, para poder cambiar los caracteres sólo una cantidad n de veces.

Por ejemplo:

frase = "pablito clavo un clavito"

Cambiar 2 veces obtenemos:

p4blit0 cl4v0 un clavito