EDA- Practica1

Begoña Cardedo.
Jonatan Perez.
Begoña Mnez de Marañon.

04/10/2013

Contenido

1 Introducción .

En esta práctica se nos pide crear una aplicación que nos permita manejar una gran cantidad de datos (miles). Para ello, se nos proporciona un fichero con los datos de unas lista de actrices y sus películas.
Con todos estos datos debemos representar por cada actriz su lista de peliculas asi como por cada película su lista de actores.
Entre las operaciones que se solicitan, podemos mencionar las siguientes:

1 - Ejecutar siempre el primero:Cargar los datos ordenados desde el fichero datos.txt.

2 - Buscar un Actor/Actriz y mostrar sus películas

3 - Añadir un Actor/Actriz ordenado y sus películas

4 - Borrar un Actor/Actriz

5 - Guardar la lista ordenada completa en el nuevo fichero datosNuevo.txt

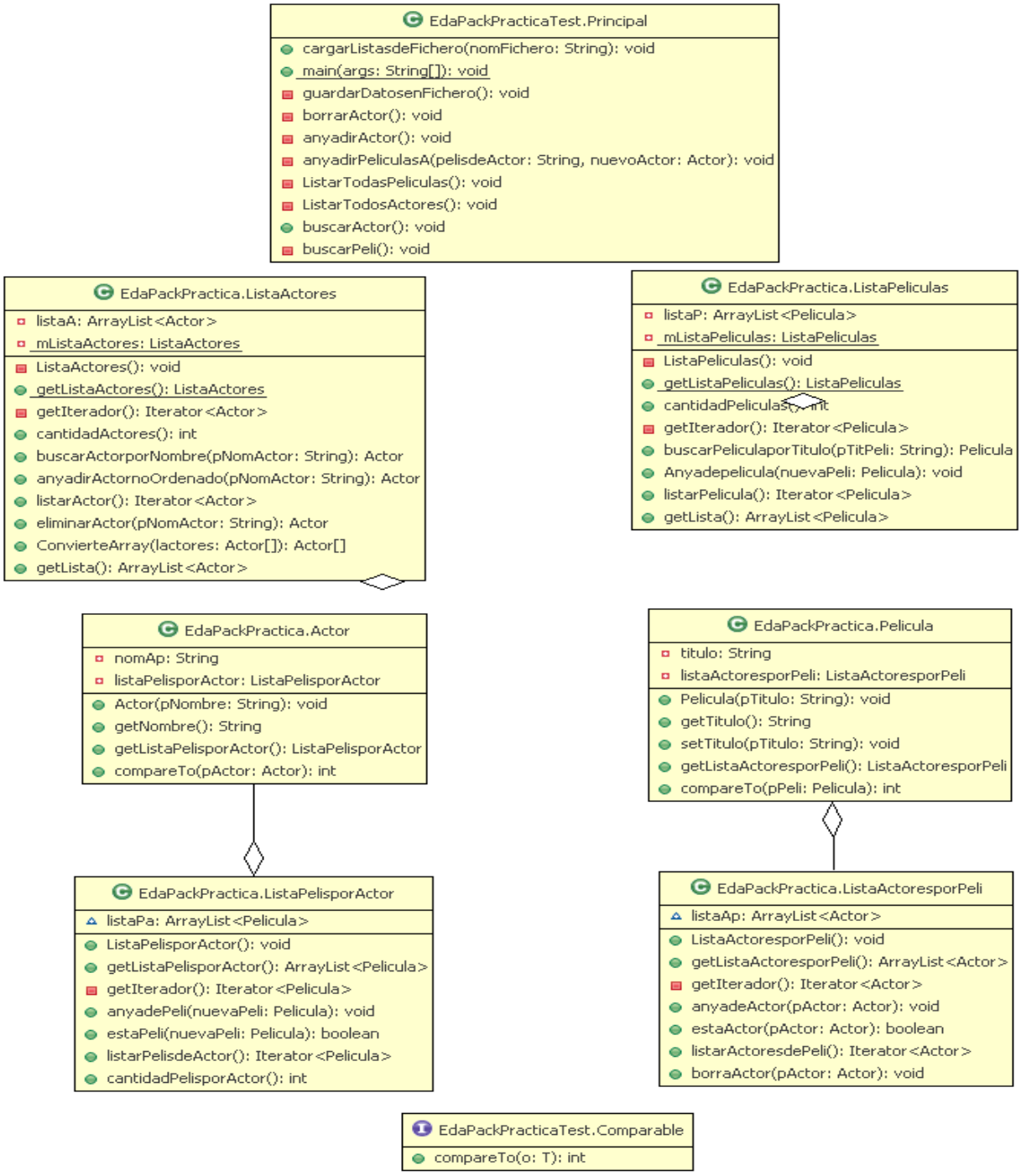
6 - Listar todos los Actores ordenados

7 - Listar todas las Películas ordenados

8 - Buscar una Película y mostrar sus actores

9 - Salir del programa

2. Diseño de las clases



3. Descripción de las estructuras de datos principales

En un principio pensamos implementar las listas con arrays, pero decidimos utlizar la clase ArrayList<T> ya que ésta gestionaba automáticamente el tamaño del array al añadir nuevos elementos, así como la introducción de nuevos elementos dándole la posición de inserción y gestionando automáticamente el desplazamiento de los elementos. Esta característica también la utilizamos para eliminar un candidato de una posición específica.

4. Diseño e implementación de las estructuras de datos principales

4.1. Método main

```
public static void main(String[] args)
{       //pre:insertar un número del menú,si se inserta un string casca.
        //post: se ejecuta el método elegido por pantalla
        Principal pr = new Principal();
        boolean salir = false;
        while(!salir){
```

Coste: el coste depende de las operaciones que el usuario quiera realizar.

4.2. Método cargarFichero

```
public void cargarListasdeFichero(String nomFichero)
        {/*pre: -el fichero se encuentra en el lugar adecuado y es no vacío, contiene los
datos adecuados.
-cada actor  tendrá ninguna o muchas películas asociadas
          post: se carga el fichero con los actores y por cada uno de ellos sus películas,si
varias aparecen en el archivo con elmismo nombre meto sólo uno,si apareciera 2 veces el mismo
actor solo lo meto una.*/
```

Casos de prueba:
- Fichero vacío.
- Fichero lleno (con datos).

Implementación:
Crear lector de clase Scanner con el fichero(renombrado actress..txt, eliminadas las 1asfilas y ultimas filas)

Hay que meter el split

Coste: el algoritmo recorre todas las líneas del fichero, para un fichero de n líneas el coste es O(n).

```
System.out.println("File loading....");
try{
Scanner entrada = new Scanner(new FileReader(nomFichero));
        String linea;
```

```
String nomPeli=null;
String nomActor=null;
ArrayList<String> tituloPelis=new ArrayList<String>();
int i=0;
while (entrada.hasNext()) {
        linea = entrada.nextLine();
        //separamos la línea por tabulaciones
        String[] datos = linea.split("\t+");.....
```

4.3. Método guardarDatosenFichero

```
private void guardarDatosenFichero() {

        System.out.println("Se va a proceder a guardar los datos actuales");
        FileWriter fichero = null;
    PrintWriter escritor = null;
    try{

            fichero = new FileWriter("datosNuevo.txt");
            escritor = new PrintWriter(fichero);
            Iterator<Actor> itrA=ListaActores.getListaActores().listarActor();
            Iterator<Pelicula> itrP;
            Actor a=null;
            int numPelis=0;
            int contA=1;
            while (itrA.hasNext())
            {

                int contador=1;
                a=itrA.next();
                if(contA!=1){
                        escritor.print("\n");
                }
                escritor.print(a.getNombre());
                itrP=a.getListaPelisporActor().listarPelisdeActor();
                numPelis=a.getListaPelisporActor().cantidadPelisporActor();
                if(numPelis!=0){
                        while (itrP.hasNext())
                        {
                                        String titPeli=itrP.next().getTitulo();
                                        if(contador>=2){

    escritor.print("\t\t\t"+titPeli+"\n");

                                        }else{.....
```

4.4. Método borrarActor

```
private void borrarActor() {
        /*pre: se recoge el nombre del actor en un formato determinado 'apellido, nombre'
,puede tener pelis asociadas o no
         post: -borro el objeto actor si existe y sus películas asociadas si tiene,no las borro
en el listado general de películas aunque se quedaran sin actor asociado.

        --busco si el actor que me dan está en listaActores devolviendo su objeto:
           -si no está saco un mensaje de que no aparece
           -si está existe:
                        *recorro las pelis asociadas del objeto actor,si tiene busco en cada
una de ellas sus actores asociados ,y borro ese actor,
```

```
                                        dejando la película que podría tener cero o muchos actores
asociados una vez borrado este actor.
                                    *borro el actor de listaActores general
                    */
                    System.out.println("Borrar:Introduzca el nombre de un Actor/Actriz de la forma
'apellido,nombre' sin espacios en blanco");
                    Scanner sc = new Scanner(System.in);
                    String nomActor = sc.nextLine();
                    Boolean esta=false;
                    Actor nuevoActor=ListaActores.getListaActores().eliminarActor(nomActor);
                    if(nuevoActor==null)……..
```

4.5. Método anyadirActor

```
private void anyadirActor() {
                    /*pre: se recoge el nombre del actor en un formato determinado 'apellido,nombre'
sin espacios en blanco y sus pelis
                     post: -si el actor no está en la lista se añade junto con sus películas
                         -si ya está le añado su/s película/s en el caso de que estén en listapelículas,si
no las creo y las añado
                    */
                    System.out.println("Añadir:Introduzca el nombre de un Actor/Actriz de la forma
'apellido, nombre'");
                    Scanner sc = new Scanner(System.in);
                    String nomActor = sc.nextLine().toString();
                    System.out.println("¿Quiere añadir también las películas en las que ha participado
ese actor? Si, No");
                     sc = new Scanner(System.in);
                    String resp = sc.nextLine();
                    Actor
nuevoActor=ListaActores.getListaActores().anyadirActornoOrdenado(nomActor);
                    Collections.sort(ListaActores.getListaActores().getLista());

                    if(resp.equalsIgnoreCase("si"))
                    {……
```

4.6 Método anyadirPeliculasA

```
private void anyadirPeliculasA(String pelisdeActor,Actor nuevoActor) {

        Pelicula
nuevaPeli=ListaPeliculas.getListaPeliculas().buscarPeliculaporTitulo(pelisdeActor);
        if(nuevaPeli==null)
                {
                        nuevaPeli=new Pelicula(pelisdeActor);
                        ListaPeliculas.getListaPeliculas().Anyadepelicula(nuevaPeli);
                        nuevaPeli.getListaActoresporPeli().anyadeActor(nuevoActor);
                        nuevoActor.getListaPelisporActor().anyadePeli(nuevaPeli);

                }else{…….
```