

Examen de programación en laboratorio de Programación I

Escuela de Ingeniería y Arquitectura
Departamento de Informática e Ingeniería de Sistemas

12 de septiembre de 2011

Tiempo: 2 horas

Este examen consta de dos ejercicios de programación. Para resolverlos se dispone de un tiempo máximo de 2 horas. La valoración del trabajo se hará de acuerdo con los siguientes criterios:

- Primer ejercicio.
 - Correcto funcionamiento del programa modificado: 4.0 puntos
 - Calidad del diseño y de la documentación: 1.0 punto
- Segundo ejercicio.
 - Correcto funcionamiento del programa modificado: 4.0 puntos
 - Calidad del diseño y de la documentación: 1.0 punto

1. Modificación del programa desarrollado en la práctica 3.^a (5.0 puntos)

Al ejecutar el método `main()` de la clase `GestorGrupo` el programa debe ofrecer una octava opción en su *Menú de Opciones*. La nueva opción debe permitir que el operador introduzca un apellido (ha de ser indiferente que lo escriba en minúsculas o mayúsculas) para que el programa presente un listado de los alumnos que tengan algún apellido (primer apellido o segundo apellido) coincidente con el introducido por el operador.

Si no hay alumnos que tengan algún apellido coincidente con el introducido por el operador entonces el programa informa de ello mediante un mensaje específico, tal como se ilustra a continuación.

No hay ningún alumno que tenga algún apellido formado por más de una palabra.

```
hendrix01:/prog1/ java -cp bin pract3.GestorGrupo
```

```
Leídos los 31 alumnos del grupo
```

MENU DE OPCIONES

=====

- 1 - Finalizar
- 2 - Listar los alumnos del grupo
- 3 - Ordenar el grupo por NIP
- 4 - Ordenar el grupo por edad
- 5 - Buscar un alumno por NIP
- 6 - Buscar un alumno por fecha de nacimiento
- 7 - Estadísticas de edades
- 8 - Buscar por apellido

Seleccione una operación [1-8]: **8**

Apellido a buscar: **Gracia**

No se ha encontrado ningún alumno con apellido Gracia

MENU DE OPCIONES

=====

- 1 - Finalizar
- 2 - Listar los alumnos del grupo
- 3 - Ordenar el grupo por NIP
- 4 - Ordenar el grupo por edad
- 5 - Buscar un alumno por NIP
- 6 - Buscar un alumno por fecha de nacimiento
- 7 - Estadísticas de edades
- 8 - Buscar por apellido

Seleccione una operación [1-8]: **8**

Apellido a buscar: **Monge**

580368 - 07/05/1992 - LORETO BERGUA MONGE
630947 - 30/08/1992 - REMEDIOS MONGE MURIEL
746198 - 12/12/1992 - MARIA ANTONIA MONGE JUANES

MENU DE OPCIONES

=====

- 1 - Finalizar
- 2 - Listar los alumnos del grupo
- 3 - Ordenar el grupo por NIP
- 4 - Ordenar el grupo por edad
- 5 - Buscar un alumno por NIP
- 6 - Buscar un alumno por fecha de nacimiento
- 7 - Estadísticas de edades
- 8 - Buscar por apellido

Seleccione una operación [1-8]: ...

Se pide modificar la clase `GestorGrupo` (y solo esa) del paquete `pract3` para que el programa presente el comportamiento modificado idéntico al que acaba de ser explicado. El código java del fichero `«GestorGrupo.java»` debe ser enviado por correo electrónico a la dirección `latre@unizar.es` ejecutando la orden

mail latre@unizar.es < GestorGrupo.java

desde el directorio donde se encuentra el fichero «GestorGrupo.java».

2. Modificación del programa desarrollado en la práctica 4.^a (5.0 puntos)

Al ejecutar el método `main()` de la clase `InfoMatriculados` el programa debe ofrecer un comportamiento diferente al especificado en la práctica 4.^a. En lugar de escribir en pantalla los NIP de los alumnos matriculados en cada asignatura, ahora deberá escribir un listado numerado de los apellidos y nombres de los alumnos. No es necesario que el listado esté ordenado de acuerdo con criterio alguno.

Para ello, puede ser de utilidad utilizar los métodos y funciones de la clase `Grupo` y `Alumno` del paquete `pract3`.

```
hendrix01:/prog1/ java -cp bin pract4.InfoMatriculados
```

```
Leídos los 31 alumnos del grupo
```

```
Código de la asignatura (0 para acabar): 30204
```

1. GARRAPIZ MADRID, MARIA TERESA
2. JUAN TOMAS, CAROLINA
3. PRIETO CARTAGENA, DIONISIO
4. RODRIGUEZ SIPAN, JORGE
5. ZAPATERO ALONSO, JACINTA
6. GALLEGO BERNARDEZ, AUGUSTO
7. ARRIZABALAGA OLIVENZA, CARLOTA
8. ANTUNEZ ZAMORA, ROLDAN
9. AZNAR POBLADOR, AZUCENA
10. MARTINEZ NAVARRO, ROLDAN
11. BERNARDEZ MOSQUERA, LIDIA
12. CANDAS ADANEZ, RUBEN
13. CARPINTERO DUQUE, FERNANDO
14. BIEL OTAL, ALONSO
15. REMACHA JUAN, IVAN
16. GALO OTANO, LUCAS
17. JULIAN MOLINS, RUBEN
18. CUERDA ESCUER, PEDRO
19. BERGUA MONGE, LORETO
20. JOSA GALO, TIBURCIO
21. BORAO BERNARDEZ, JACINTO
22. MATEU FRANCES, ESCOLASTICO
23. CARPINTERO GUTIERREZ, ELENA
24. MERINO ARAGON, JOSE LUIS
25. MONGE JUANES, MARIA ANTONIA

Código de la asignatura (0 para acabar): **30207**

1. GARRAPIZ MADRID, MARIA TERESA
2. JUAN TOMAS, CAROLINA
3. SISAMON OLIVITO, JUAN IGNACIO
4. RODRIGUEZ SIPAN, JORGE
5. GALLEGO BERNARDEZ, AUGUSTO
6. ARRIZABALAGA OLIVENZA, CARLOTA
7. ANTUNEZ ZAMORA, ROLDAN
8. AZNAR POBLADOR, AZUCENA
9. MARTINEZ NAVARRO, ROLDAN
10. BERNARDEZ MOSQUERA, LIDIA
11. CANDAS ADANEZ, RUBEN
12. BIEL OTAL, ALONSO
13. ABADIA AZANZA, JUAN IGNACIO
14. REMACHA JUAN, IVAN
15. GALO OTANO, LUCAS
16. JULIAN MOLINS, RUBEN
17. CUERDA ESCUER, PEDRO
18. BERGUA MONGE, LORETO
19. MONGE MURIEL, REMEDIOS
20. JOSA GALO, TIBURCIO
21. BORAO BERNARDEZ, JACINTO
22. MATEU FRANCES, ESCOLASTICO
23. MERINO ARAGON, JOSE LUIS

Código de la asignatura (0 para acabar): **31247**

** Error: no hay ninguna asignatura cuyo código sea el 31247

Código de la asignatura (0 para acabar): ...

Se pide modificar la clase `InfoMatriculados` (y solo esa) del paquete `pract4` para que el programa presente un comportamiento idéntico al que acaba de ser explicado. El código Java del fichero «`InfoMatriculados.java`» debe ser enviado por correo electrónico a la dirección `latre@unizar.es` ejecutando la orden:

```
mail latre@unizar.es < InfoMatriculados.java
```

desde el directorio donde se encuentra el fichero «`InfoMatriculados.java`».