

Programación

UD11 – Gestión Dinámica de Memoria

Ejercicios 1

1. Utiliza como fuente de datos el fichero DATA.csv disponible en Moodle (UD08) para llevar a cabo este ejercicio. Crea la clase Registro que corresponda con la misma estructura de los registros del fichero DATA.csv. Crea un programa que lea la información del fichero, cree un registro por cada línea del fichero (excepto la cabecera) y añada cada registro a una lista. Utilizando los métodos de ordenación y búsqueda de las listas el programa deberá mostrar:
 - Los registros del país Francia (FR).
 - Los registros con email de yahoo ordenados por nombre.
 - Los registros con el último login entre noviembre de 2015 y enero de 2017 ambos incluidos ordenados por fecha de manera descendente.
 - Los nombres y apellidos de los registros cuyo nombre (first_name) empiecen por 'A' ordenados por fecha (descendente) del último inicio de sesión (last_login).
2. Crea un programa que elija al azar 3 números distintos entre 0 y 9.
3. Crea un programa que elija al azar 50 números entre 0 y 99. Después el programa pedirá al usuario introducir un número entre 0 y 99 y le dirá cuántas veces aparece dicho número en la lista.
4. Escribe un programa que permita crear una lista de palabras y que, a continuación, pida dos palabras y sustituya la primera por la segunda en la lista.

Ejemplo:

Lista inicial: "monitor", "keyboard", "trackpad", "computer"

Sustituir la palabra:

"trackpad"

Por:

"mouse"

Lista final: "monitor", "keyboard", "mouse", "computer"

5. Escribe un programa que permita crear una lista de palabras y que, a continuación, pida una palabra al usuario y la elimine de la lista.

Ejemplo:

Lista inicial: "monitor", "keyboard", "trackpad", "computer"

Eliminar la palabra:

"trackpad"

Lista final: "monitor", "keyboard", "computer"

6. Escribe un programa que crear una lista de 30 números aleatorios entre 0 y 99. A continuación, el programa mostrará todos los números, pero sin elementos repetidos en caso de que los hubiera.
7. Escribe un programa que permita crear dos listas de palabras y que, a continuación, escriba las siguientes listas (en las que no debe haber repeticiones):
 - Lista de palabras que aparecen en las dos listas.
 - Lista de palabras que aparecen en la primera lista, pero no en la segunda.
 - Lista de palabras que aparecen en la segunda lista, pero no en la primera.
 - Lista de palabras que aparecen en alguna de las dos listas.

Ejemplo de listas:

```
var list1 = new List<string>() {"Ann", "Carol", "Mary", "Joe",  
"John", "Kevin", "Jack", "Jack", "Lucy", "Laura", "George",  
"Ann", "Johnny", "Charlie", "Hillary"};
```

```
var list2 = new List<string>() { "Ann", "Martha", "Noel",  
"Mark", "Pamela", "Elisabeth", "Laura", "Jane", "Mildred",  
"Laura", "Rue", "Ann", "Jacob", "Sidney", "Sophia", "Tyler" };
```

8. Crea un programa que pida al usuario introducir una frase y muestre las letras ordenadas en orden alfabético junto con el número de veces que aparece dicha letra en la frase.