

Práctica Final: El juego de las letras

Andrés Herrera Poyatos

14/01/2015

Este es el código de la Práctica Final de la asignatura Estructura de Datos de segundo del Doble Grado de Matemáticas e Ingeniería Informática. De encontrar algún error o requerir información al respecto mi e-mail es andreshp9@gmail.com

Estructura del Código

Se proporciona una carpeta **letras** con la siguiente estructura de directorios:

- **bin** : Carpeta donde se almacenan los archivos ejecutables. Se encuentra vacía a la espera de ejecutar el *Makefile*.
- **doc** : Carpeta con la documentación generada por *Doxygen*. Contiene únicamente la configuración de *Doxygen* a la espera de ejecutar la orden `make documentacion`.
- **obj** : Carpeta donde se almacenan los módulos objeto. Se encuentra vacía a la espera de crear los módulos objeto con el *Makefile*.
- **include** : Carpeta con los archivos de cabecera (*.h*) y el código de las clases template (*.cpp*). Contiene los siguientes archivos:
GeneralTree.h, GeneralTree.cpp, Trie.h, Trie.cpp, Dictionary.h, Letter.h, LetterBag.h, LetterSet.h
- **src** : Carpeta con el código fuente (*.cpp*). Contiene los siguientes archivos:
Dictionary.cpp, LetterBag.cpp, LetterSet.cpp, main.cpp

El *Makefile* proporcionado genera un *.o* por cada *.cpp* en **src** con el mismo nombre. Estos módulos objeto se compilan juntos para dar lugar al ejecutable **letras**.

Para una mayor información sobre las clases ver la documentación de *Doxygen*.

Uso del Programa

En primer lugar se debe obtener el ejecutable con el **Makefile**. La sintaxis para el uso del ejecutable responde a la siguiente estructura genérica:

```
./bin/letras <argumentos opcionales> <fichero con el diccionario> <argumentos pertinentes>
```

El ejecutable **letras** admite diferentes opciones como primer argumento. Según la opción elegida cambia considerablemente los posteriores argumentos del programa. Es por tanto recomendable prestar atención al subapartado **opciones**.

Opciones

- **Por defecto (no requiere elegir opción)**: Juega al juego de las letras para los ajustes dados. Sintaxis:

```
./bin/letras <Diccionario> <Archivo con las Letras> <Nº de letras>
```

- **-a:** Añade al diccionario seleccionado las palabras dadas. Sintaxis:

```
./bin/letras -a <Diccionario> <Varias Palabras>
```

- **-b:** Busca en el diccionario las palabras dadas como argumento e indica si existen. Sintaxis:

```
./bin/letras -b <Diccionario> <Palabras a buscar>
```

- **-c:** Calcula un archivo con las letras de un diccionario dado, sus frecuencias y una puntuación entre 1 y 10 asignada según estas frecuencias. En caso de no dar el último argumento el archivo creado se muestra por pantalla. Sintaxis:

```
./bin/letras -c <Diccionario> <Opcional: Nombre del archivo donde almacenar el resultado>
```

- **-e:** Elimina del diccionario seleccionado varias palabras. Sintaxis:

```
./bin/letras -e <Diccionario> <Varias Palabras>
```

- **-h:** Se muestra por pantalla un manual de ayuda con las diferentes opciones del programa. Sintaxis:

```
./bin/letras -h
```

- **-l:** Muestra por pantalla aquellas palabras del diccionario con longitud deseada. Sintaxis:

```
./bin/letras -l <Diccionario> <Longitud>
```

- **-p:** Juega al juego de las letras en su versión por puntuación. Sintaxis:

```
./bin/letras -p <Diccionario> <Archivo con las Letras> <Nº de letras>
```

- **-u:** Une los dos diccionarios seleccionados almacenando el diccionario resultante con el nombre dado. En el caso de no proporcionarse un nombre para el nuevo diccionario se muestra su contenido por pantalla. Sintaxis:

```
./bin/letras -u <Diccionario 1> <Diccionario 2> <Opcional: Nombre del nuevo diccionario>
```