

DOCUMENTACIÓN – EJERCICIO ANAGRAMA

MP LAB22 - Esteban Pérez Cabral

Descripción funcional

El algoritmo *Anagrama* se encarga de la lectura de un fichero de texto, éste se indica por la entrada estándar (teclado), que se encarga de analizar dicho texto. Primero, analiza la primera palabra de dicho texto, a partir de ésta las compara las siguientes palabras con la primera, observando si son anagramas o permutaciones. Así sucesivamente hasta analizar el texto completo, indicando en pantalla el número de permutaciones de dicha palabra.

Descripción modular

Basándonos en el análisis del algoritmo *Anagrama* ya hecho en la parte teórica, no requiere un nuevo análisis de este. Pero si que indicaremos como implementamos en lenguaje C cada módulo:

- Módulo *palabra*: implementado en “*Palabra.h*” y “*Palabra.c*”. En el primero trae la declaración de las estructuras y tipos de datos, y los prototipos de las funciones externas. En el último incluye las implementaciones de todas las funciones, así como su especificación.
- Módulo *texto*: en “*Texto.h*” incluye la declaración tipo de datos y los prototipos de las funciones visibles. Además en “*Texto.c*” incluye la implementación y su especificación de cada función.
- Módulo *permutaciones*: similar a los anteriores, “*Permutaciones.h*” incluye la cabecera de las funciones externas, y en “*Permutaciones.c*” incluye implementación y especificación.

Instalación y Ejecución

Los archivos del código fuente se incluirán para una posible modificación de cualquier usuario, pero para ejecutar el programa sólo es necesario copiar el archivo “*Anagrama.exe*” en cualquier directorio de su ordenador y abrirlo. Añadir como información adicional que el fichero de texto a analizar debe estar en el mismo directorio. En caso contrario, debe indicarse la ruta del fichero.

Por ejemplo, si el fichero se llama “*textoprueba.txt*”, al abrir el programa, en cualquier caso, introduciremos por entrada “*C:\ruta\textoprueba.txt*”, si lo tenemos en la misma carpeta bastaría con introducir “*textoprueba.txt*”.

Además de indicar el fichero, el usuario no tiene que interactuar más para la ejecución del programa, sólo esperar la impresión del resultado, o en su defecto, la indicación de algún error que ocurriera durante la ejecución.

Prueba - Caja negra

Función “*longitud*”:

Palabra vacía: perfecto, ningún resultado inesperado.

Palabra con 4 caracteres: bien.

Función “*palabra_es_permut*”:

Introduciendo pares de palabras diferentes;

Flora Larof: resultado esperado.

Flora Florá: presenta un error, ya que no reconoce adecuadamente el acento.

Vasos Casa: correcto.

Casa Niño: correcto.

46&%~ Hola: lo esperado.

Prueba - Caja blanca

Para el programa completo:

Fichero vacío: bien.

Fichero con una palabra: resultado correcto

Fichero con varias palabras: resultado esperado.