

Programación III

Proyecto Segundo Parcial

Word Counter Code

Objetivo:

Ejercitar y afianzar el uso de estructuras de datos mediante una aplicación concisa y de uso general.

Formas de presentación:

Exposición y defensa del programa e informe con código fuente y comentarios.

Forma de trabajo:

Grupos de no más de 2 personas.

Criterios de Evaluación:

- Compilación correcta del Programa.
- Ejecución del Programa.
- Eficiencia del Programa.
- Exposición y defensa del trabajo.
- Preguntas sobre las estructuras utilizadas y a cada integrante.
- Esquemas Gráficos Borradores.
- Legibilidad del Código.
- Comentarios del Código.
- El cumplimiento de lo especificado en el enunciado del ejercicio. El enunciado especifica lo mínimo a alcanzar para obtener el máximo puntaje.
- Que la lógica de programación usada considere estrechamente el uso de las estructuras de datos enseñadas en clases y que las estructuras elegidas sean las más convenientes para resolver el problema. La Eficacia será considerada. La eficiencia no, al menos en esta instancia.
- La consideración de los casos extremos y su abordaje con excepciones. Usar directivas Try, Throw y Catch.
- Tiempo de entrega.

Enunciado:

Realizar una aplicación informática que realice las funciones básicas del comando “wc” de Linux y las mejoras planteadas a continuación:

1. Las funciones básicas son:

- a. contar letras
- b. contar palabras
- c. contar líneas

2. Indicar cuantas palabras diferentes hay en el texto.

3. Mostrar las palabras diferentes en orden alfabético.

4. Mostrar las palabras diferentes en orden de más ocurrencias. El usuario puede informar un grupo de palabras a ignorar mediante otro archivo de texto.

Fecha Presentación Proyecto Parcial: 13/10/202

5. Informar la cantidad de veces que aparece una o un grupo de palabras solicitadas por el usuario.

Manual de uso:

Uso normal:

wce file.txt

Argumentos posibles:

- **-palabras [n]:** Mostrará las n primeras palabras en orden alfabético. Si n no es ingresado o vale 0, mostrará todas las palabras.
- **-ocurrencias [n]:** Mostrará las n palabras y la cantidad de ocurrencias de c/u ordenadas de mayor a menor por ocurrencia. Si n no es ingresado o vale 0, mostrará todas las palabras
- **-mostrar [palabra],[palabra] :** Mostrará la o las palabras pasadas como argumento ordenadas por ocurrencia.
- **-excluir [palabra],[palabra] :** Modifica los comandos ocurrencias y palabras haciendo que no muestren las palabras pasadas como argumentos.
- **-excluirf [archivo.txt] :** Modifica los comandos ocurrencias y palabras haciendo que no muestren las palabras que contiene el archivo.txt.

Ejemplo:

// file.txt

el veloz murciélago hindú
comía feliz cardillo y kiwi.
La cigüeña tocaba el saxofón
detrás del palenque de paja

wce file.txt

lineas: 4

palabras: 19

letras: 119

palabras diferentes: 18

wce -palabras 3 file.txt

cardillo

cigüeña

comía

wce -mostrar el file.txt

palabras:

el 2

wce -mostrar “el, cardillo” file.txt

palabras:

2 el

1 cardillo

wce - ocurrencias file.txt

2 el

1 cardillo

1 cigüeña

...

wce - ocurrencias - excluir cardillo file.txt

2 el

1 cigüeña

...

wce - ocurrencias - excluirf ign.txt file.txt

2 el

1 cigüeña

...