



Apellido y Nombres	Legajo	# de Hojas

Parte práctica. Enunciado.

Se desea desarrollar una aplicación para moderar texto. La aplicación recibe dos argumentos por línea de comandos obligatorios:

- Primer argumento: Es una cadena con el texto completo a moderar.
- Segundo argumento: Es una cadena con las palabras que componen la lista de palabras no permitidas.

La aplicación deberá:

1. Validar los argumentos de entrada. No ejecuta si no hay una opción válida en cada uno de los argumentos. Mostrará el mensaje de uso por **stderr**, y terminará con código de error.
2. Tomará el texto ingresado en el primer argumento (ver Pistas donde se indica como ejecutar el programa), y reemplazará por una 'x' cada carácter de cada palabra que coincida con alguna de las que figuran en la lista de palabras no permitidas.
3. Finalizado el análisis deberá presentar por pantalla, por cada palabra de la lista de palabras no permitidas, la cantidad de veces que esta apareció en el texto moderado, a razón de una por línea

Condiciones

1. El punto 3 del requerimiento es deseable (incrementa la calificación final del examen siempre que los ítems 1 y 2 estén resueltos correctamente).
2. No se permite utilizar las funciones de la **libc** definidas en **string.h**
3. Debe utilizar no menos de tres archivos fuentes: Un header **modera.h**, un archivo **main.c**, en donde esté la parte principal del programa, un archivo **my_string.c**, donde estén las funciones de manejo de cadenas, y **aux.c** para el resto de las funciones auxiliares que requiera su código.
4. **main.c** debe tener un comentario en su encabezado con las instrucciones para generar el ejecutable llamado **modera**.

Pistas:

El programa cuyo ejecutable se debe llamar **modera** se invocará pasando el contenido de dos archivos como argumentos. Los archivos están en el directorio **publico** de git.

```
./modera ``cat texto.txt`` ``cat lista.txt``
```

Puede usar el script **run.sh** provisto en el mismo directorio (ejecutar ./run.sh en su directorio de trabajo)

Para trabajar con cadenas se sugiere fuertemente desarrollar funciones equivalentes a las de libc definidas en string.h.

En este link las tiene agrupadas:

https://www.tutorialspoint.com/c_standard_library/string_h.htm



Apellido y Nombres	Legajo	# de Hojas

Teoría

- Convertir los siguientes números a binario de 16 bits de tamaño en Ca2, y signo y magnitud.
 - 4977
 - 1231
- ¿Cual es el contenido de un archivo header?. ¿En donde se encuentra el código objeto de las funciones de biblioteca?
- La variable a definida e inicializada según la sentencia: `int * a = &b;`, ocupa la dirección 0x2000 y su contenido es 0x3000. Completar la tabla siguiente:

¿Que tipo debe tener la variable b?	
Valor de c definida como <code>int **c = &a;</code>	
Resultado de <code>printf ("%p\n", a);</code>	
Completar para que el resultado sea 0x3000 usando c	<code>printf ("%__\n", __);</code>
Resultado de <code>printf ("%x\n", *c);</code>	

- Cual de las siguientes representaciones corresponde a un dato de tipo int almacenado en memoria en una CPU que trabaja en Little endian, cuyo valor es 0x 6290A71F

Dirección	a	b	c	d	e
0x10000000	0x62	0x90	0xA7	0x1F	Ninguna de las anteriores
0x10000001	0x90	0x62	0x1F	0xA7	
0x10000002	0xA7	0x1F	0x62	0x90	
0x10000003	0x1F	0xA7	0x90	0x62	

- ¿Por donde pasa(n) el/los argumento(s) y donde existe(n) la(s) variable(s) que utiliza una función invocada desde otra?. Dibuje gráficamente la zona de memoria ubicando cada uno de los argumentos, variables, y otros items que corresponda.