

Realice un archivo “.c”, que posea las funciones que se detallan más abajo. Éstas deben estar correctamente documentadas, indicando los parámetros que recibe, que devuelve y características particulares si correspondiese – se recomienda utilizar Doxygen para realizar esta documentación.

Todas las funciones deben iniciar con el prefijo “xtr\_”

En paralelo, realice un programa (en otro archivo “.c”) que utilice estas funciones a modo de prueba y corrobore su correcto funcionamiento.

**1.-** Realice una función que reciba un string y devuelva la cantidad de caracteres que posee.

```
int xtr_len (const char *);
```

**2.-** Realice una función que reciba un string y un carácter e informe la cantidad de veces que éste se repite en el string.

```
int xtr_qtty (const char *, char);
```

**3.-** Realice una función que reciba un string y convierta todos sus caracteres a mayúscula, a minúscula o que cada palabra inicie con mayúscula y continúe con minúscula.

```
Char * xtr_chg_case (char *, int);
```

**4.-** Realice una función que reciba dos strings y determine si son iguales o no.

```
int xtr_cmp (const char *, const char *);
```

Si ambos son iguales, debe retornar cero, en caso de ser distintos devolverá un valor mayor a cero si el primer string es alfabéticamente mayor al segundo y devolverá un valor negativo en caso contrario.

**5.-** Realizar funciones en base a la del ejercicio anterior de manera que:

**5.a.-** Se pueda indicar la cantidad de caracteres que se quiere comparar como máximo.

**5.b.-** No sea case-sensitive. Es decir que las mayúsculas y minúsculas sean tratadas como iguales.

**6.-** Realice una función que copie un string en otro. A su vez, permita definir la cantidad de caracteres máximo a copiar o que se copie todo.

En caso de copia parcial, debe agregar al string copiado el carácter de terminación de string.

Queda a criterio del desarrollador el valor de retorno de la función.

**7.-** Realizar una función que busque un carácter dentro de un string. La función debe retornar en caso de encontrarlo, la dirección de la primer coincidencia.

**8.-** Ídem al anterior pero devolviendo la última coincidencia.

Nota: para los puntos 7 y 8, como obtiene a partir del valor retornado la posición relativa dentro del array (índice)

**9.-** Realizar una función que busque un string dentro de otro string. La función debe retornar en caso de encontrarlo, la dirección del primer carácter de la primer coincidencia.