

EXAMEN PRÁCTICAS
ARQUITECTURA DE COMPUTADORES (RISC-V)
1º Ingeniería de Robótica Software, URJC

Fuenlabrada, 18 de Junio de 2019

AVISO: Asegúrate que tus programas **cumplen** con los siguientes criterios. Si no se cumple alguno de ellos la **nota máxima** de tu examen será de **2 puntos**

- **Cumplimiento de especificaciones.** Se deben cumplir las especificaciones indicadas en el enunciado: nombres de funciones, nombres de archivos, funcionalidad, número de parámetros, etc. Compruébalo antes de entregar el examen
- **Respetar el convenio.** Resuelve las preguntas **sin violar** el convenio del uso de registros
- **Sin errores en tiempo de ejecución** (Runtime errors). Tus programas no deben generar excepciones al ejecutarse

Necesitamos crear la función `void upper(char *pstr)` que recibe como argumento un **puntero** a una **cadena de caracteres** y convierte esa cadena a **mayúsculas**. Los caracteres que ya estuviesen en mayúsculas se quedan igual, así como el resto de caracteres que NO son letras. Por tanto, sólo hay que convertir los caracteres desde la 'a' a la 'z' en mayúsculas, dejando el resto como estuviesen

Por ejemplo, la cadena `"Summer is coming**-2019\n"` se convertirá a `"**SUMMER IS COMING**-2019\n"` al aplicar esta función (Recuerda que el carácter '\n' es el salto de línea)

Para su implementación, el jefe de proyecto ha determinado que se debe llamar a la función `void upper_char(char *c)`, la cual realiza el paso a mayúsculas de **un único carácter**, cuyo puntero se pasa como parámetro (c)

Se pide implementar las siguientes funciones y los programas principales indicados:

1. **Función upper_char** (2 ptos): `void upper_char(char *c)`: Pasar a mayúsculas el carácter apuntado por c. Para pasar un carácter a mayúsculas basta con restarle 32 a su código ASCII. Esta operación sólo se realiza con los caracteres de la 'a' (código ascii 97) a la 'z' (código ascii 122). El resto de caracteres se deja como estaban. Esta función se debe implementar en el **fichero upper_char.asm**
2. **Programa principal** de prueba de **upper_char** (1 pto). Dentro del **fichero upper_char.asm** se debe incluir un programa principal que compruebe que la función `upper_char` funciona correctamente. Deberá definirse una cadena en tiempo de compilación, imprimirla en la consola, llamar a la función `upper_char` para convertir a mayúsculas **el primer carácter**, imprimir la nueva cadena y terminar

3. **Función upper** (4 ptos): *void upper(char *pstr)*. Convertir a mayúsculas la cadena apuntada por el puntero pasado como parámetro. Se debe usar un **algoritmo iterativo** que recorra la cadena, convirtiendo cada uno de los caracteres a mayúsculas **llamando a la función upper_char**. Se debe implementar en el **fichero upper.asm**
4. **Programa principal** de prueba de upper (1 pto). Dentro del **fichero upper.asm** se debe incluir un programa principal que compruebe que la función upper funciona correctamente. Deberá definirse la siguiente cadena en tiempo de compilación :*"Summer is coming**-2019\n"*, imprimirla en la consola, convertirla a mayúsculas llamando a upper, imprimirla en la consola y terminar
5. **Fichero upper2.asm** (2 ptos). Ampliar el programa anterior para que además de la conversión anterior, pida al usuario una cadena en tiempo de ejecución, la imprima en mayúsculas en la consola y termine. Se debe implementar en el **fichero upper2.asm** (Por tanto, este programa hace dos conversiones a mayúsculas, una de la cadena definida en tiempo de compilación, y otra con la cadena introducida por el usuario en tiempo de ejecución)