

CC3301 Programación de software de sistemas – Tarea 5 – Otoño 2024 – Profesores Mateu/Ibarra/Urrea

La función *sort* está programada en C en el archivo *sort-c.c* y en assembler Risc-V en *sort-rv.s*. Esta función ordena lexicográficamente un arreglo de strings usando un algoritmo ridículamente ineficiente, porque toma tiempo $O(n^3)$. En *sort-rv.s*, el código equivalente en C está comentado, mostrando la ubicación de las variables en los registros.

El encabezado de la función es: `void sort(char *a[], int n);`

En el arreglo de strings *a* vienen *n* frases como "*Gonzalez DIEGO*".

El archivo *sort-rv-im.s* es una copia de *sort-rv.s* y *sort-c-im.c* es una copia de *sort-c.c*. Modifique las funciones *sort* en *sort-c-im.c* y en *sort-rv-im.s* de modo que se ordene el arreglo **ignorando diferencias entre mayúsculas y minúsculas**. La siguiente tabla muestra el ordenamiento lexicográfico versus el mismo ignorando diferencias entre mayúsculas y minúsculas:

Orden lexicográfico	Ignorando diferencias entre mayúsculas/minúsculas
FERNANDEZ maria	FERNANDEZ maria
Gonzalez DIEGO	fernandez monica
Gonzalez peDro	fernandez veri
PEREZ JOSE	gonzalez Diego
fernandez monica	Gonzalez DIEGO
fernandez veri	Gonzalez peDro
gonzalez Diego	PEREZ JOSE
perezZ juan	perez josefa
perez josefa	perezZ juan

Observe que en el orden lexicográfico las mayúsculas son menores que las minúsculas.

Instrucciones

Baje *t5.zip* de U-cursos y descomprímalo. Contiene el *Makefile* y los archivos que necesita para hacer esta tarea. Ejecute el comando *make* sin parámetros para recibir instrucciones sobre la ubicación de los archivos que debe modificar y cómo compilar, ejecutar y depurar. En particular lea los tips para la depuración y la solución de problemas.

Restricciones

No hay restricciones para la programación de *sort-c-im.c*. Para la programación de *sort-rv-im.s* **Ud. solo puede modificar el código que compara los elementos consecutivos**. Está claramente delimitado en el archivo original. No modifique nada más. Sin esta restricción la tarea sería trivial. **Además para hacer la comparación no puede invocar otras funciones**. Una vez hecha la comparación, la ejecución debe

continuar en la etiqueta *.decision* y el resultado de la comparación debe quedar en el registro *t1*. Si $t1 > 0$, los números en $p[0]$ y $p[1]$ están desordenados y por lo tanto se intercambiarán. Si $t1 \leq 0$ no se intercambiarán.

Ayuda

- Modifique en *sort-c-im.c* la función que compara strings de manera que se ignore la diferencia entre mayúsculas y minúsculas. Haga un esfuerzo en llegar a la función más pequeña, porque así será menor la cantidad de líneas en assembler que deberá programar y depurar. Pruebe *sort-c-im.c* ejecutando el comando: *make sort-c-im.run* (*make sort-c-im.ddd* para depurar)
- Una vez que pase exitosamente la prueba de *sort-c-im.c*, ejecute el comando *make sort-c-im.s* y luego estudie en el archivo *sort-c-im.s* la traducción a assembler Risc-V de la función que compara strings. Use un código similar para comparar los strings en la zona delimitada en *sort-rv-im.s* y así completar su tarea. Recuerde: sin invocar otras funciones. Pruebe su tarea con: *make sort-rv-im.run* (*make sort-rv-im.ddd* para depurar)
- Revise si el toolchain para compilar y ejecutar programas para Risc-V está instalado en */opt/riscv*. Si no encuentra ese directorio, descargue el archivo *riscv.tgz* (*riscv-arm.tgz* para Arm) de [esta carpeta de google drive](#) e instale el *toolchain* para RiscV con estos comandos:

```
cd ... directorio en donde descargó riscv.tgz ...
sudo bash
cat riscv.tgz | ( cd / ; tar zxvf - )
```
- En la clase auxiliar del viernes 10 de mayo se estudió la solución de una tarea similar de un semestre pasado.

Entrega

Entregue por medio de U-cursos el archivo ***im.zip*** generado con el comando *make zip*. Este comando verifica que su tarea funcione correctamente y luego genera *im.zip* con los archivos *sort-c-im.c*, *sort-rv-im.s* y *resultados.txt* con la ejecución de la tarea. **Recuerde descargar de u-cursos lo que entregó, descargar nuevamente los archivos adjuntos** y vuelva a probar la tarea tal cual como la entregó. Esto es para evitar que Ud. reciba un 1.0 en su tarea porque entregó los archivos equivocados. Créame, sucede a menudo por ahorrarse esta verificación. Su tarea debe ordenar correctamente, si no será rechazada. Se descontará medio punto por día de atraso (excluyendo sábados, domingos, festivos o vacaciones).