

66:20 Organización de computadoras

Trabajo práctico 1: Conjunto de instrucciones MIPS.

1. Objetivos

Familiarizarse con las herramientas de software que usaremos en los siguientes trabajos, implementando un programa (y su correspondiente documentación) que resuelva el problema piloto que presentaremos más abajo.

2. Alcance

Este trabajo práctico es de elaboración grupal, evaluación individual, y de carácter obligatorio para todos alumnos del curso.

3. Requisitos

El trabajo deberá ser entregado personalmente, en la fecha estipulada, con una carátula que contenga los datos completos de todos los integrantes.

Además, es necesario que el trabajo práctico incluya (entre otras cosas, ver sección 7), la presentación de los resultados obtenidos, explicando, cuando corresponda, con fundamentos reales, las causas o razones de cada resultado obtenido.

El informe deberá respetar el modelo de referencia que se encuentra en el grupo¹, y se valorarán aquellos escritos usando la herramienta $\text{T}_{\text{E}}\text{X}$ / $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$.

4. Recursos

Usaremos el programa GXemul [1] para simular el entorno de desarrollo que utilizaremos en este y otros trabajos prácticos, una máquina MIPS corriendo una versión reciente del sistema operativo NetBSD [2].

En la clase del 30/8 hemos repasado, brevemente, los pasos necesarios para la instalación y configuración del entorno de desarrollo

También debe respetarse la convención de llamadas a funciones provista por la cátedra.

5. Descripción.

Al igual que el anterior trabajo práctico, se debe implementar una versión minimalista del comando *sort*[4], aunque solo se debe implementar el algoritmo *Stooge sort*[3].

¹<http://groups.yahoo.com/group/orga6620>

6. Implementación.

El programa debe leer los datos de entrada desde *stdin* o bien desde uno o mas archivos. La salida del programa debe imprimirse por *stdout*, mientras que los errores deben imprimirse por *stderr*. La implementación del algoritmo Stooge sort debe ser realizada en assembly MIPS32, respetando el siguiente encabezado:

```
void stoogesort(void *array, unsigned int i, unsigned int j)
```

El resto del programa (carga de datos, impresión de resultados, etc debe ser escrito en lenguaje C). El ejecutable entregado debe respetar la convención de llamadas presentada en clase.

7. Informe.

El informe debe ser entregado en un folio transparente y debe incluir:

- Informe describiendo el desarrollo del trabajo práctico y conclusiones pertinentes.
- Comando(s) para compilar el programa.
- Corridas de prueba, con los comentarios pertinentes.
- CD conteniendo todo el material digital, etiquetado con los nombres de los integrantes del grupo.
- Código fuente.
- Diagramas de stack de cada función implementada en MIPS32.
- Este enunciado.

8. Fechas de entrega.

- Primera entrega: 25-10
- Vencimiento: 8-11

Referencias

- [1] GXemul, <http://gavare.se/gxemul/>.
- [2] The NetBSD project, <http://www.netbsd.org/>.
- [3] Stoogesort http://en.wikipedia.org/wiki/Stooge_sort
- [4] sort [http://en.wikipedia.org/wiki/Sort_\(Unix\)](http://en.wikipedia.org/wiki/Sort_(Unix))