

Laboratorio 3: Arquitectura y Organización de Computadores

Profesor: Viktor Tapia

Ayudante de cátedra: Mauricio Cortes y Muryel Constanzo

Ayudante de Tarea: Vicente Alevear y Luciano Yevenes

5 de junio 2024

1. Reglas Generales

Para la siguiente tarea se debe utilizar la plataforma QtARMSim¹ para diseñar un programa que cumpla con los requerimientos de la sección 2. Se exigirá que el código se presente de la forma más limpia y ordenada posible. Deberá incluir un README con la identificación de los estudiantes que desarrollaron la tarea, además de cualquier supuesto utilizado.

2. Programas a implementar

2.1. Contexto

Después del "Gran Desastre en el Certamen de Arquitectura 2024", la cátedra ha decidido ofrecer una oportunidad de redención. El Sistema de Asistencia y Mejora (SAM) ha diseñado un conjunto de desafíos que pondrán a prueba sus habilidades de programación en ARM Assembly. ¡Prepárense para demostrar su dominio de los algoritmos y la arquitectura de computadores!

2.2. Optimizar el Pipeline de paquetes

En un esfuerzo por optimizar el procesamiento de datos en una cadena de producción, se ha implementado un pipeline (esto se puede entender como fila, cadena, línea, etc) de paquetes. Cada paquete representa una unidad de datos procesados, y se debe decidir la mejor manera de recolectar estos datos para maximizar el rendimiento del sistema. Las reglas del pipeline son estrictas:

Escribe un programa en ARM Assembly que reciba como entrada una lista de paquetes y devuelva la mayor cantidad de datos que se puede obtener de esa lista siguiendo la regla: puedes procesar un paquete y saltarte los siguientes dos, o puedes pasar al siguiente paquete.

Entrada:

- Una lista con los valores de los paquetes (solo valores enteros positivos).

Salida:

- La mayor cantidad de data posible.

¹ <https://pypi.org/project/qtarmsim/>

Entrada:

1 2 3 4 5

Salida:

7

Ejemplo 1:

Para el ejemplo anterior se saltó el primer paquete paso al segundo, escogió este y por **regla de las Pipeline** salto el paquete 3 y 4 pudiendo solo seleccionar el 5, obteniendo el 7 de la salida.

Entrada:

3 5 1 5 2

Salida:

8

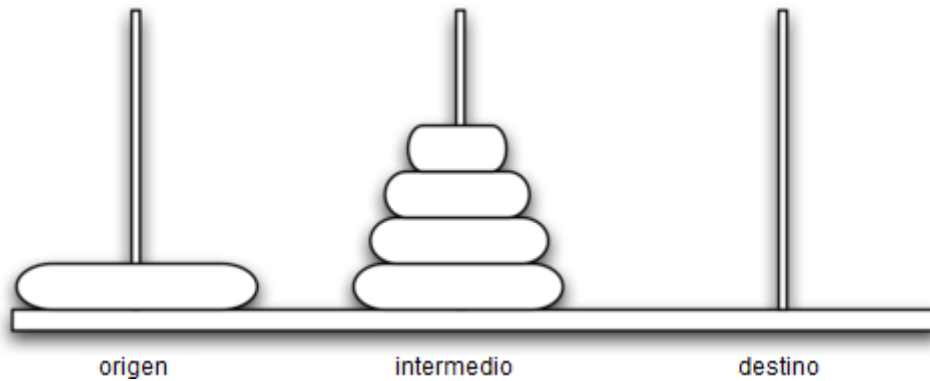
Ejemplo 2:

Para el ejemplo anterior selecciono el primer y por **regla de las Pipeline** salto el paquete segundo y tercer paquete seleccionar el 5 o saltar este y seleccionar el 2, pero el mayor se selecciona el 5 por ser mayor, obteniendo el 8 de la salida.

2.3. La Torre de Hanoi en el Laboratorio de Procesadores

El segundo desafío consiste en resolver el problema clásico de la Torre de Hanoi, una tarea que simboliza la transferencia de datos ordenada y eficiente en un sistema de múltiples procesadores. Este rompecabezas no solo es una prueba de habilidad lógica, sino que también representa la construcción de una secuencia de operaciones eficiente.

Escribe un programa en ARM Assembly que resuelva el problema de la Torre de Hanoi de la manera más eficiente posible para un número determinado de discos. El programa debe recibir como entrada el número de discos y **mostrar la cantidad de movimientos realizados para resolver el rompecabezas** en la consola de QtARMSim.



Entrada:

- Un número entero positivo representando el número de discos.

Salida:

- La cantidad de movimientos realizados debe mostrarse en la consola de QtARMSim.

Ejemplo 1:

- *Entrada*

```

8
9      .data
10
11 Entrada:  .word 3
12

```

- *Salida*

```

LCD
El numero de movimientos es
7

```

Ejemplo 2:

- *Entrada*

```

8
9      .data
10
11 Entrada:  .word 2
12

```

- *Salida*

```

LCD
El numero de movimientos es
3

```

2.4. Requerimientos:

- Utiliza las interfaces stdin (.data) y stdout (consola) de QtARMSim para la entrada y salida de datos.
- **Se deben respetar la regla de las Pipeline.**

- El programa del primer desafío debe funcionar correctamente para cualquier lista de paquetes con un tamaño máximo de 10 elementos.
- El programa del segundo desafío debe funcionar correctamente para cualquier número de discos entre 1 y 5 haciendo la mínima cantidad posible de movimientos.

3. Readme

Debe contener como mínimo:

- Nombre, Rol y Paralelo de los integrantes.
- Especificación de los algoritmos y desarrollo realizado.
- Supuestos utilizados

4. Consideraciones

- Se deberá trabajar de a pares. Se deberá entregar en Aula a más tardar el día 22/05/2024 a las 23:59 horas. Se descontarán 5 puntos por cada hora o fracción de atraso. Las copias serán evaluadas con nota 0 en el promedio de las tareas.
- La tarea debe realizarse en QtARMSim. Se recomienda que se familiarice rápidamente con la plataforma, y ante cualquier duda consulte con sus compañeros o directamente con los ayudantes lo antes posible. El único responsable si no acude a alguien para resolver sus dudas a tiempo es usted.
- La entrega considera de uno a 2 programas según consideren pertinente, pero se deberá respetar lo dicho anteriormente en los requerimientos, todo esto junto con el README. Los archivos deberán ser comprimidos y enviados juntos en un archivo .zip de nombre LAB3_ROL1_ROL2.
- Si no se entrega README, o si su programa no funciona, la nota es 0 hasta la corrección.
- Una vez entregadas las notas de la tarea existirá un plazo de 5 días para apelar. Transcurrido este plazo las notas no podrán ser modificadas.
- Acá se les deja un enlace para que puedan usar un simulador para las torres de Hanói <https://www.geogebra.org/m/NqyWJVra>