Práctica 1: Torres de Hanói

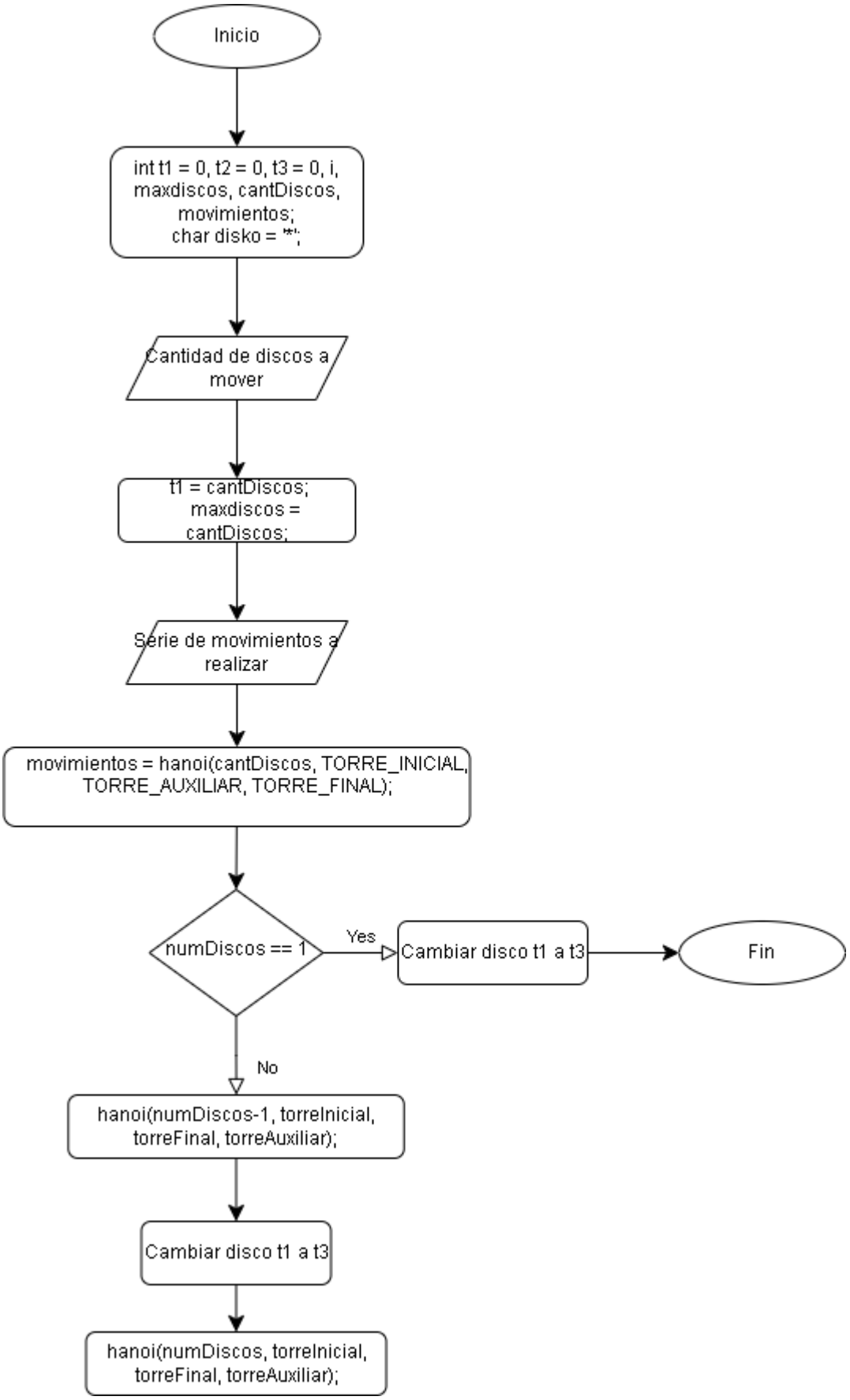
ITESO

Universidad Jesuita
de Guadalajara

Organización y Arquitectura de Computadoras
25 de octubre 2023

Profesor:
Juan Pablo Ibarra Esparza
Equipo:
Luis Raúl Acosta Mendoza 739199
Yochabel Martínez Cázares 738438

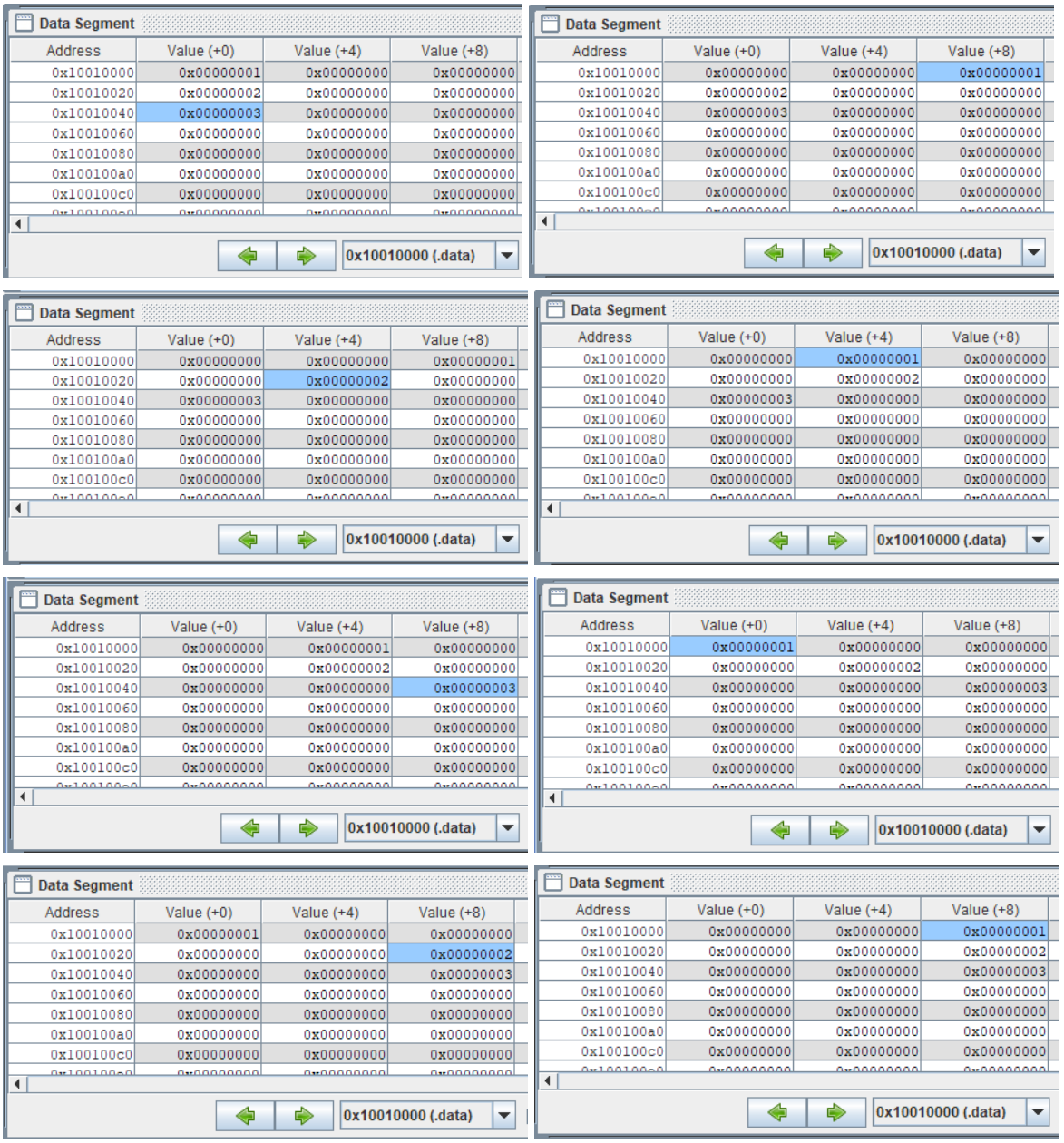
Diagrama de flujo



Decisiones de diseño

- Para una visualización más fácil, decidimos colocar las torres en el segmento data de forma vertical. Por lo que la torre 1 comienza colocando el disco 1 en la dirección 0x10010000, el resto de discos se colocan de forma incremental dando saltos de 0x20 (32) bytes para cada nivel. La torre 2 comienza en 0x10010004 y la torre 3 en 0x10010008.
- Los discos se mueven de una torre a otra solo en su nivel correspondiente. Por ejemplo, el disco 2 solo se puede mover del nivel 2 de su torre actual al nivel 2 de la torre destino.

Simulación en RARS con 3 discos



Análisis de comportamiento del stack

Se invoca hanoi con s0 = 3 y entra a hanoirecursive ya que s0 != 1
Primera llamada recursiva:

- Push al stack, se decrementa s0 a 2 y se intercambian torre auxiliar y torre final.
- Se vuelve a llamar hanoi
- Como s0 != 1, vuelve a entrar a hanoirecursive

Segunda llamada recursiva:

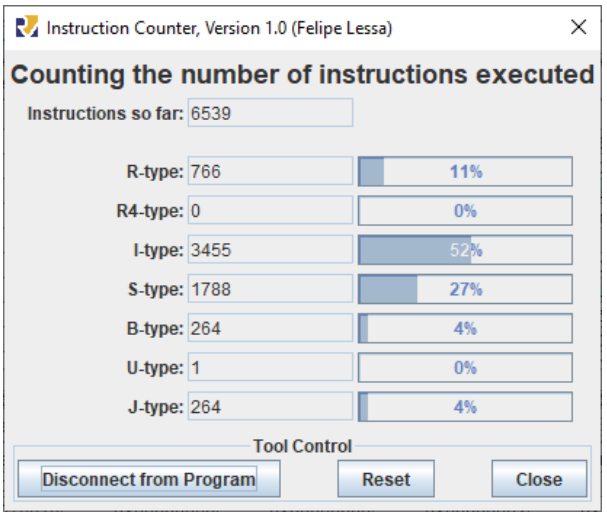
- Push al stack, se decrementa s0 a 1 y se intercambia torre auxiliar y torre final.
- Se vuelve a llamar hanoi
- Como s0 == 1, se mueve un disco de la torre inicial a la final y se regresa.
- Pop del stack, se mueve un disco de la torre inicial a la final y se vuelve a hacer una 2da llamada recursiva con torre inicial y auxiliar intercambiados.

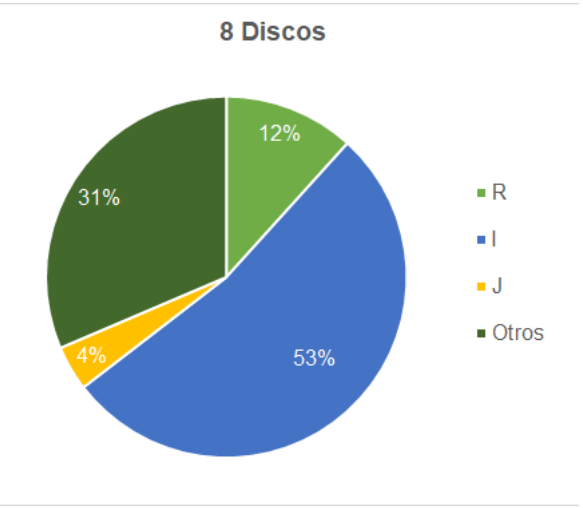
Tercera llamada recursiva:

- Mismo procedimiento con la 2da llamada recursiva, solo que este tiene los roles intercambiados.

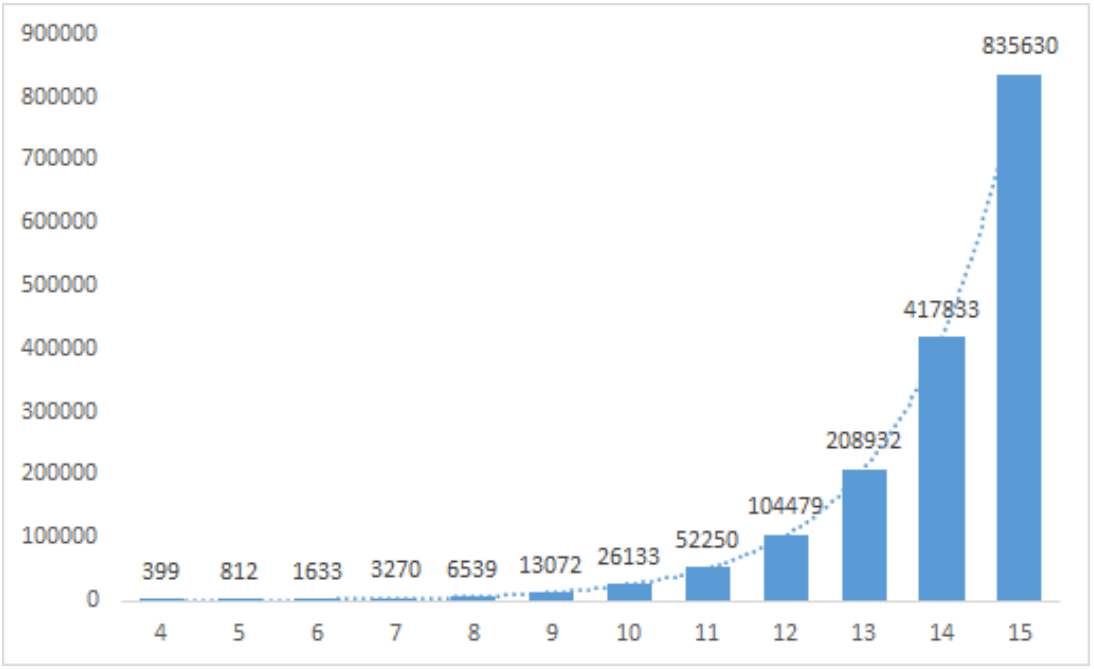
En el stack se almacenan los valores mencionados para cada vez que se llama a hanoi de forma consecutiva. Una vez que se alcanza la condición de que s0 == 1, es cuando se leen los datos del stack en orden inverso al que se ingresaron.

Instruction Count R, I y J con 8 discos





Instruction Count con 4 hasta 15 discos



Conclusiones

Luis: Fue interesante realizar esta práctica, me sirvió para reforzar el cómo se piensa para programar en ensamblador. La parte más difícil considero que fue el movimiento de discos, inicialmente, nuestro plan fue mover un disco del tope de una torre al tope de otra torre, lo que implica tener que encontrar el tope de las torres primero. Lo resolvimos cambiando el diseño al que está ahora especificado más arriba, puesto que funciona de forma más fácil y directa.

Yochabel: Esta práctica fue una buena opción ya que se pudo desarrollar todo lo que es traducir un código C a ensamblador. En lo personal siento que la cuestión de las llamadas recursivas fue lo que más se me dificulta traducir, pero en esta práctica fue una manera de desarrollarlo más, el único problema que se tuvo al momento de realizarlo fue el mover los discos de un principio ya que llegaba un momento en el que se iban a direcciones equivocadas, pero al final se pudo resolver.