



IIC2343 – Arquitectura de Computadores (II/2016)

Tarea 4

Fecha de entrega: miércoles 12 de octubre de 2016 a las 23:59 horas

Ayudante encargado: Patricio Díaz (pndiaz1@uc.cl)

Programación en assembly x86

En esta tarea continuarán profundizando en la programación de *bajo nivel* de un computador. Utilizando el assembly x86-16 visto en clases, deberán escribir un programa que dado un un grafo acíclico dirigido¹, encuentre un orden topológico² de este. Al igual que en la tarea anterior, todos los datos requeridos se entregarán al programa mediante utilizando los labels *size*, *graph*, *ordering*, todos de tipo **db**. En particular, el grafo se debe codificar utilizando una matriz de adyacencia³, que debe ser almacenada en orden de filas y comenzar en la dirección de memoria asociada al label *graph*. El resultado del algoritmo (el orden topológico) debe indicarse a partir de la dirección asociada al label *ordering*, utilizando un arreglo de tamaño **size**. Es importante tener en consideración que los nombres de los nodos del grafo **parten desde 1 y no desde 0**. Un ejemplo de declaración de los labels *size*, *graph*, *ordering* puede verse a continuación:

```
JMP main
size      db  4
graph     db  0,1,1,0,0,0,0,1,0,0,0,1,0,0,0,0
ordering  db  0,0,0,0
main:
.
```

Para probar la tarea, se debe utilizar el emulador emu8086, disponible en <http://www.emu8086.com/>.

Entrega y evaluación

La tarea se debe realizar de **manera individual** y la entrega se realizará mediante un cuestionario a través del sitio del curso. El formato de entrega debe consistir en un único archivo .asm, que lleve como nombre el número de alumno, y que contenga el código fuente. No incluya en este archivo contenido relacionado con el entorno de desarrollo utilizado. El no cumplir este formato de entrega implicará un descuento de **1.0** punto en la nota final de la tarea.

¹https://en.wikipedia.org/wiki/Directed_acyclic_graph

²https://en.wikipedia.org/wiki/Topological_sorting

³https://en.wikipedia.org/wiki/Adjacency_matrix

La evaluación se realizará con posibles notas **7.0**, **4.0** ó **1.0**, donde un **7.0** corresponde a una tarea sin errores, un **4.0** a una tarea con pequeños errores y un **1.0** a cualquier otro caso. Tareas incompletas serán evaluadas con nota **1.0**. En caso de atraso, se aplicará un descuento de **1.0** punto por cada 12 horas o fracción.

Finalmente, todas las tareas serán analizadas electrónicamente por posibles copias o plagio (copia de código no trivial desde otras fuentes, sin incluir referencia a esta). En caso que se detecte alguna situación de este tipo, el castigo corresponde a un **1.0** en la tarea y la modificación del criterio del cálculo del promedio de las tareas, eliminándose ahora la mejor nota en vez de la peor. En caso de una segunda detección de copia, el o los alumnos afectados reprobarán inmediatamente el curso con nota **1.1** y la situación será informada a la Dirección de Docencia de la Escuela de Ingeniería.