



IIC2343 – Arquitectura de Computadores (II/2017)

Enunciado Tarea 04

Memoria caché, 50 % de la entrega 04

Fecha de entrega: 19 de Noviembre del 2017, 23:59 hrs.

1. Descripción

En esta tarea tendrán la oportunidad de experimentar con el diseño de una memoria caché, desde el punto de los mecanismos de acceso a datos y las políticas de escritura. Más específicamente, deberán programar la simulación de una jerarquía de memoria de dos niveles (caché y memoria principal), la cual se ve enfrentada a una serie de peticiones de lectura y escritura.

La tarea desarrollada debe recibir como entrada a través de la consola el nombre de un archivo de texto que contiene la especificación completa de la jerarquía requerida y de la secuencia de lecturas/escrituras. Como salida, la tarea debe imprimir el *hit-rate* y el *write-miss* correspondientes a la secuencia, donde el *hit-rate* se define como la cantidad de **accesos realizados a la caché** (lectura y escritura) dividida por la cantidad de **accesos totales** indicadas en la secuencia y el *write-miss* se define como la cantidad de **escrituras realizadas en la memoria principal** dividida por la cantidad de **escrituras totales** indicadas en la secuencia.

El archivo de entrada tendrá un formato **propiedad=valor(es)**, donde **propiedad** indica el elemento a configurar y **valor** indica el o los valores para el elemento. En particular, las propiedades a configurar y sus posibles valores se indican a continuación:

- **CACHE SIZE**: indica el tamaño de la memoria caché en función de la cantidad de palabras (bytes).
- **LINE SIZE**: indica el tamaño de las líneas de la memoria caché en función de la cantidad de palabras (bytes).
- **MEM SIZE**: indica el tamaño de la memoria principal en función de la cantidad de palabras (bytes).
- **CORR**: indica la función de correspondencia a utilizar, que puede ser una de las siguientes: DM (*mapeo directo*), FA (*fully associative*), 2W (*2-way associative*).
- **SUBS**: indica la política de sustitución a utilizar, que puede ser una de las siguientes: FIFO, LRU, LFU.
- **WRITE**: indica la política de escritura a utilizar, que puede ser una de las siguientes: WT (*write-through*), WB (*write-back*).
- **SEC**: indica una secuencia ordenada de largo arbitrario de lecturas/escrituras. Los valores serán números que indican la dirección de memoria requerida, seguidos de la letra W en caso de tratarse de una escritura.

Un ejemplo de un posible archivo de entrada puede verse a continuación:

```
CACHE_SIZE=8
LINE_SIZE=2
MEM_SIZE=32
CORR=FA
SUBS=FIFO
WRITE=WB
SEC=0,1,2,3W,4,5,6,7,8,9,8,7W,6,5,4,3,2,1,0W
```

Puede tomar las siguientes consideraciones:

- El *input* solo serán valores definidos anteriormente. En ningún momento se ingresarán números o datos que no corresponden a la propiedad definida.
- Los accesos a memoria siempre serán números entre 0 y MAX_SIZE - 1.
- El orden de las propiedades **puede no** ser igual al del ejemplo. Por ejemplo puede partir indicando SEC, luego WRITE y después los demás atributos.
- No hay espacios entre los acceso de memoria de SEC o entre alguna propiedad y su valor.
- **No deben** aproximar ningún valor.

2. Implementación

Para la realización de esta tarea deberán trabajar en el lenguaje Python 3.x, sin ocupar bibliotecas externas que implementan la lógica de la tarea. Está permitido el uso de bibliotecas para leer archivos, imprimir en pantalla y manejar parámetros por línea de comando. Para el uso de alguna librería que no sea para lo indicado anteriormente, deben hacer una *issue* donde pregunten por la librería y se explique el motivo de su uso.

3. Corrección

Para ejecutar la tarea y corregirla, se deberá entregar por línea de comandos el nombre del archivo que contiene los números:

```
$ python tarea.py nombre_archivo.txt
```

Luego de esto, el programa deberá imprimir en pantalla los resultados requeridos. No se evaluará código en esta ocasión, solo que el *output* sea el indicado (con similitud de 2 decimales) y que no se use ninguna librería prohibida.

Los test que usaremos serán con las siguientes combinaciones:

1. CORR = DM, SUBS=FIFO, WRITE=WB
2. CORR = DM, SUBS=FIFO, WRITE=WT
3. CORR = DM, SUBS=LRU, WRITE=WB
4. CORR = DM, SUBS=LRU, WRITE=WT
5. CORR = DM, SUBS=LFU, WRITE=WB

6. CORR = FA, SUBS=FIFO, WRITE=WB
7. CORR = FA, SUBS=FIFO, WRITE=WT
8. CORR = FA, SUBS=LRU, WRITE=WB
9. CORR = FA, SUBS=LRU, WRITE=WT
10. CORR = FA, SUBS=LFU, WRITE=WB
11. CORR = 2W, SUBS=FIFO, WRITE=WB
12. CORR = 2W, SUBS=FIFO, WRITE=WT
13. CORR = 2W, SUBS=LRU, WRITE=WB
14. CORR = 2W, SUBS=LRU, WRITE=WT
15. CORR = 2W, SUBS=LFU, WRITE=WB

Lo que puede variar en cada test es CACHE SIZE, LINE SIZE y MEM SIZE.

Cada test valdrá 4 décimas.

4. Contacto

- Francesca Lucchini: flucchini@uc.cl
- Felipe Pezoa: fipezoa@uc.cl
- Hernán Valdivieso: hfvaldivieso@uc.cl
- Luis Leiva: lileiva@uc.cl

5. Integridad académica

Los alumnos de la Escuela de Ingeniería de la Pontificia Universidad Católica de Chile deben mantener un comportamiento acorde a la Declaración de Principios de la Universidad. En particular, se espera que mantengan altos estándares de honestidad académica. Cualquier acto deshonesto o fraude académico está prohibido; los alumnos que incurran en este tipo de acciones se exponen a un Procedimiento Sumario. Es responsabilidad de cada alumno conocer y respetar el documento sobre Integridad Académica publicado por la Dirección de Docencia de la Escuela de Ingeniería.

Específicamente, para los cursos del Departamento de Ciencia de la Computación, rige obligatoriamente la siguiente política de integridad académica. Todo trabajo presentado por un alumno para los efectos de la evaluación de un curso debe ser hecho individualmente por el alumno, sin apoyo en material de terceros. Por “trabajo” se entiende en general las interrogaciones escritas, las tareas de programación u otras, los trabajos de laboratorio, los proyectos, el examen, entre otros. Si un alumno copia un trabajo, obtendrá nota final 1,1 en el curso y se solicitará a la Dirección de Docencia de la Escuela de Ingeniería que no le permita retirar el curso de la carga académica semestral. Por “copia” se entiende incluir en el trabajo presentado como propio partes hechas por otra persona.

Obviamente, está permitido usar material disponible públicamente, por ejemplo, libros o contenidos tomados de Internet, siempre y cuando se incluya la referencia correspondiente.

Lo anterior se entiende como complemento al Reglamento del Alumno de la Pontificia Universidad Católica de Chile. Por ello, es posible pedir a la Universidad la aplicación de sanciones adicionales especificadas en dicho reglamento.