



Tarea 3

Aspectos preliminares

El objetivo de esta evaluación es que profundice en los aspectos prácticos y teóricos de los tópicos del capítulo 5 del curso. En esta evaluación hay preguntas teóricas y de programación, que estarán marcadas con **(T)** o una **(P)** al lado del número que las identifica, para diferenciarlas claramente.

La evaluación esta diseñada para que deba buscar material y recursos que no están en el sitio del curso para contestar las preguntas. Tomando esto en consideración, como esta tarea tiene un fin pedagógico, la regla de integridad académica que debe cumplir es la siguiente:

Toda respuesta a cualquier pregunta debe ser desarrollada y escrita individualmente. Esto significa que al momento de editar la respuesta, no se debe estar mirando o usando un texto escrito por otra persona, un video hecho por otra persona, o cualquier material que haya sido desarrollado por otra persona. Tampoco es válido memorizar una respuesta obtenida de otra fuente y luego escribirla. En otras palabras, lo que sea que se escriba debe provenir directamente de usted. Sin embargo, antes de comenzar a escribir cada respuesta puede estudiar y comentar la pregunta con otros alumnos (no la respuesta, solo la pregunta), leer libros, mirar videos, o realizar cualquier otra acción que le ayude a responder la pregunta. Esta regla aplica tanto a pregunta teóricas como prácticas.

Finalmente, la entrega deberá incluir un archivo llamado INTEGRIDAD.txt, en donde afirma que ha leído, entendido y respetado la regla de arriba. De no cumplirse esto, la tarea no será corregida y obtendrá la nota mínima.

Entrega

La fecha de entrega es el viernes 23/10 hasta las 23:59. El formulario para la entrega se encuentra en la siguiente dirección <https://forms.gle/z5g7b8migun4vZP39>. En este formulario encontrará la descripción del formato de entrega para cada una de las preguntas. Tenga en consideración que una vez vencido el plazo de entrega, las respuestas solo pueden ser entregadas 1 vez (usando los días de gracia).

1. Jerarquía de memoria y memoria caché

La nueva versión del emulador incluye la funcionalidad de exportar todos los accesos a memoria realizados por un programa, indicando la memoria y orden en que fueron hechos. Esta exportación se realiza una vez que termina de ejecutarse un programa y se encuentra en formato JSON. En base a esto, las siguientes preguntas requerirán que programe un algoritmo simple en assembly del computador básico y luego, utilizando la secuencia de accesos, que simule el comportamiento de distintos tipos de memoria caché.

- a) (P) Escriba en assembly del computador básico un programa que dada una matriz de número enteros de tamaño $M \times N$, calcule la norma ℓ_1 de esta, es decir, la suma de los valores absolutos de los coeficientes. En caso que la norma sea mayor a 255, trunque el resultado a la capacidad de los registros. **(1.0 pto.)**
- b) (P) Ejecute el programa desarrollado en el ítem anterior y genere los accesos a memoria. Luego, escriba un programa en Python que calcule el hit-rate de esta secuencia en una jerarquía de memoria con las siguientes características:

- i. Memorias caché separadas para instrucciones y datos, cada una de 16 palabras, líneas de 4 palabras, mapeo directo. **(1.0 pto.)**
 - ii. Memorias caché separadas para instrucciones y datos, cada una de 16 palabras, líneas de 4 palabras, fully associative, sustitución Bélády. **(1.0 pto.)**
 - iii. Memorias caché separadas para instrucciones y datos, cada una de 16 palabras, líneas de 4 palabras, 2-way associative, sustitución LRU. **(1.0 pto.)**
 - iv. Memoria caché unified de 16 palabras de 8 bits, líneas de 4 palabras, fully associative, sustitución random. **(1.0 pto.)**
- c) (P) Modifique el programa escrito en el primer ítem de esta pregunta, sin cambiar su lógica de funcionamiento, de manera que tenga en promedio peor hit-rate que la versión original. Evalúe el hit-rate usando las memorias caché del ítem anterior. **(1.0 pto.)**

2. Multiprogramación

En esta pregunta también utilizará la nueva versión del emulador con el fin de generar los accesos a memoria de programas en assembly, pero a diferencia de la pregunta anterior, acá utilizará los accesos para simular el comportamiento de un sistema multiproceso. Antes de empezar esta pregunta, deberá elegir N programas en assembly del computador básicos, ya sea tomados desde los apuntes o de ejercicios que ud. haya resuelto anteriormente (no considere programas que interactúen con dispositivos de I/O), y generar sus accesos a memoria. La cantidad de programas debe ser tal que $N \geq 5$ y que en conjunto los N programas utilicen al menos tres veces el tamaño de la memoria de datos del computador básico.

- a) (P) Escriba un programa en Python que calcule el número de faltas de página generadas por accesos a la memoria de datos, luego de ejecutar todos los programas hasta que finalicen. Para esto, asuma páginas de 16 palabras y que cada proceso se ejecuta por 10 ciclos antes de ser sustituido por otro. Tanto el orden de ejecución de los procesos como el criterio de mapeo de páginas a marcos deber ser definido por ud. **(4.0 ptos.)**
- b) (T) ¿Qué pasa con el contenido de la memoria caché de un sistema multiproceso al haber un cambio de contexto? ¿Cómo conviene manejar el contenido de esta desde el punto de vista del hit-rate? **(2.0 ptos.)**

Política de Integridad Académica

Los alumnos de la Escuela de Ingeniería deben mantener un comportamiento acorde al Código de Honor de la Universidad:

“Como miembro de la comunidad de la Pontificia Universidad Católica de Chile me comprometo a respetar los principios y normativas que la rigen. Asimismo, prometo actuar con rectitud y honestidad en las relaciones con los demás integrantes de la comunidad y en la realización de todo trabajo, particularmente en aquellas actividades vinculadas a la docencia, el aprendizaje y la creación, difusión y transferencia del conocimiento. Además, velaré por la integridad de las personas y cuidaré los bienes 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.