



IIC2343 – Arquitectura de Computadores (II/2018)

Tarea 10

Fecha de entrega: jueves 22 de noviembre de 2018 a las 23:59 horas

Pregunta 1

- a) Describa detalladamente qué pasa si se intenta ejecutar un archivo ejecutable binario del computador básico, en un computador **x86** de 16 bits. **(1 pto.)**
- b) Un *disassembler* es un programa que transforma código binario ejecutable en *assembly*. Describa como funcionaría un *disassembler* para el computador básico. ¿Es posible obtener el *assembly* original a partir de un programa ubicado en la memoria de instrucciones? **(2 ptos.)**
- c) Describa detalladamente un mecanismo para generar un archivo ejecutable en **x86**, a partir del resultado de un *assembler* para el computador básico. Describa claramente cómo trabaja cada una de las transformaciones intermedias. **(3 ptos.)**

Pregunta 2

Una imagen en escala de grises se define por su tamaño en píxeles y la profundidad de cada uno de estos, *i.e.* la cantidad de valores distintos que puede tomar un pixel. En base a esto, conteste las siguientes preguntas:

- a) Dada una memoria con bus de direccionamiento y palabras de 1 byte, ¿cómo almacenaría en ella una imagen en escala de grises de 5×5 píxeles y 16 bits de profundidad? **(0.75 ptos.)**
- b) Obtenga una expresión para la dirección de memoria del elemento (i, j) de la imagen del ejercicio anterior, donde $(1, 1)$ corresponde a las coordenadas del pixel superior izquierdo. Asumiendo que la imagen está almacenada en la memoria a partir de la dirección **0x6E**, calcule la dirección de memoria del pixel ubicado en $(3, 4)$. **(0.75 ptos.)**
- c) Una imagen en colores se diferencia de una imagen en escala de grises en que cada uno de los píxeles posee ahora 3 valores de intensidad distintos, uno para cada color (**RGB**). Esto es análogo a tener 3 imágenes de escala de grises distintas, donde cada una describe la distribución de intensidad para un color. ¿Como almacenaría una imagen en colores de 7×7 píxeles y 1 byte de profundidad por color en una memoria igual a la descrita anteriormente? **(1.5 ptos.)**
- d) ¿Por qué no sirve esta misma memoria para almacenar una imagen en color de 7×7 píxeles y 2 bytes de profundidad por color? Indique el problema y una posible solución. **(1 pto.)**
- e) Obtenga una expresión para la dirección de memoria del elemento (i, j, k) de una imagen en colores, donde i, j es la ubicación del pixel y k es uno de los tres canales de color. **(2 ptos.)**

Pregunta 3

- a) Indique para qué se utilizan bits de paridad en los computadores. **(0.5 ptos.)**
- b) Imagine que tiene un computador **x86** cuya memoria posee bit de paridad para cada palabra y detecta un error en la lectura. ¿Qué podría hacer el computador frente a este caso? **(1 pto.)**
- c) ¿Qué modificaciones tendría que hacer al computador básico para que funcione con bit de paridad sobre *Input/Output*? Sea lo más detallado posible. **(3 ptos.)**
- d) Para una comunicación que utiliza bit de paridad entre un computador y un I/O: ¿Qué factor limita la velocidad de transferencia? **(1.5 ptos.)**

Pregunta 4

De acuerdo a la documentación de **Java**, una implementación de la **JVM** (*Java Virtual Machine*) debe implementar una máquina de *stack* virtual de 32-bits que ejecuta el *bytecode* de Java que produce el compilador. Además, la **JVM** debe administrar el espacio de memoria de datos que utilizan los procesos e implementar alguna forma de liberar direcciones que ya no se utilizan.

Al respecto, indique:

- a) ¿Qué ventajas le entrega este diseño respecto a una máquina de registros? **(1 pto.)**
- b) ¿Qué desventajas produce esto de cara al usuario? **(2 ptos.)**
- c) ¿Es posible crear un *chip* que cumpla las especificaciones de la JVM? ¿Por qué? **(3 ptos.)**

Pregunta 5

Se desea modificar el computador básico a nivel de microarquitectura e ISA. Para los siguientes puntos detalle las modificaciones que haría. Debe utilizar diagramas de componentes, conexiones y tablas de *opcodes* e instrucciones cuando corresponda.

- a) Permitir al computador la ejecución de dos operaciones aritméticas-lógicas iguales, pero con distintos argumentos, de manera simultánea, *i.e.* el proceso debe tomar solo un ciclo del *clock*. **(1.5 ptos.)**
- b) Permitir la autoprogramabilidad manteniendo memorias de datos e instrucciones separadas. **(1.5 ptos.)**
- c) **(Solo microarquitectura)** Agregar un registro que cumpla las mismas funciones que el BP de la arquitectura x86. **(1.5 ptos.)**
- d) **(Solo microarquitectura)** Agregar una FPU y dos registros dedicados solo a operaciones de punto flotante. Los resultados de la FPU deben poder almacenarse tanto en estos registros como en la memoria de datos. **(1.5 ptos.)**

Entrega y evaluación

La tarea se debe realizar de **manera individual** y la entrega se realizará a través de GitHub. El formato de entrega debe consistir en un archivo llamado **tarea10.tex**, junto con los recursos necesarios para que este pueda ser compilado correctamente. Archivos que no compilen o no cumplan este formato de entrega implicará nota **1.0** en la tarea. En caso de atraso, se aplicará un descuento de **1.0** punto por cada 6 horas o fracción.

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. 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 (grupo) para los efectos de la evaluación de un curso debe ser hecho individualmente por el alumno (grupo), 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 (grupo) copia un trabajo, los antecedentes serán enviados a la Dirección de Docencia de la Escuela de Ingeniería para evaluar posteriores sanciones en conjunto con la Universidad, las que pueden incluir reprobación del curso y un procedimiento sumario. Por “copia” se entiende incluir en el trabajo presentado como propio partes hechas por otra persona. Está permitido usar material disponible públicamente, por ejemplo, libros o contenidos tomados de Internet, siempre y cuando se incluya la cita correspondiente.