

## Instrucciones

- La tarea es individual, debe resolverla e investigar por su propia cuenta. Cualquier intento de plagio se procesará de acuerdo al reglamento de la Universidad de Costa Rica.
- La fecha de entrega es el día lunes 02 de junio de 2018 a las 23:55. Entregables: Debe entregar en el sitio virtual del curso, un único archivo con extensión .s, el archivo debe llamarse <carne>\_tarea<número de tarea>\_grupo<número de grupo>.s, (ejemplo B12345\_tarea1\_grupo1.s) además el archivo debe contener un pequeño encabezado con sus datos y explicación breve del código implementado.
- Es sumamente necesario que el código contenga comentarios que explican el porqué de lo realizado.
- La función main debe contener únicamente llamadas a otras funciones y no grandes bloques de código.
- Su código debe ser ejecutable en QTSpm.

## Memoria Cache

En esta tarea se deberá implementar un controlador de cache en el lenguaje de programación Mips.

- La cache se diseñará para una memoria principal de 4GB, el tamaño de la cache es de 512KB asociativa en 4 vías, el tamaño de bloque es de 64B y la política de reemplazo es LRU.
- La cache estará compuesta por dos array, uno de conjuntos (será del tamaño del número de bloques por vía de la cache) y otro de etiquetas (será del tamaño de la cantidad de bloques que puede almacenar la cache).
- Cada entrada del array de conjuntos será de tamaño de una palabra, de esta palabra se utilizará cada byte para almacenar el estado de la estructura LRU. Para direccionar este array solo se requiere saber el número del conjunto.
- Cada entrada del array de etiquetas será del tamaño de una palabra, en esta se almacenará, partiendo del bit menos significativo hacia el más significativo, el bit de válido, el dirty bit, la etiqueta. Para direccionar esta cache se deberá sumar a la base del array el número de conjunto multiplicado por 16 y a esto sumarle 0, 4, 8 o 12 para seleccionar la posición de cache dentro del conjunto. Dibuje la memoria para entender esto.
- Como entrada para la cache utilice los datos provistos en el archivo accesses.s adjunto, se le recomienda probar antes con una entrada más pequeña, note que el array accesses acaba en 0 (NULL).
- Al final de la ejecución el programa deberá imprimir la tasa de fallos.

## Opcional

- Implemente la política RRIP y compare los resultados. 20p