

LABORATORIO 2 ORGANIZACIÓN DE COMPUTADORES: CACHÉ

HERNÁN OLMEDO

Profesores:

Felipe Garay,

Erika Rosas,

Nicolás Hidalgo

Ayudante:

Ian Mejias

TABLA DE CONTENIDOS

ÍNDICE DE FIGURAS.....	iv
ÍNDICE DE CUADROS	v
CAPÍTULO 1. INTRODUCCIÓN.....	7
CAPÍTULO 2. MARCO TEÓRICO.....	9
2.1 ETAPAS DEL PIPELINE	9
2.2 HAZARDS	9
CAPÍTULO 3. DESARROLLO	11
CAPÍTULO 4. DISCUSIONES DE LOS RESULTADOS	13
CAPÍTULO 5. CONCLUSIÓN.....	15
CAPÍTULO 6. BIBLIOGRAFÍA.....	17

ÍNDICE DE FIGURAS

ÍNDICE DE CUADROS

CAPÍTULO 1. INTRODUCCIÓN

Los computadores hoy en día juegan un papel preponderante en el desarrollo tecnológico y pese a su rápida evolución, las exigencias no se quedan atrás con el paso del tiempo y cada vez son más las áreas que exigen procesar grandes cantidades de datos en un tiempo reducido. Por ejemplo procesar los datos de una sonda espacial, áreas como las finanzas, la edición de gráficos, entre otras. Una de las partes claves en la rapidez de respuesta de un computador es el manejo de la memoria, en este informe trataremos la memoria de acceso rápido llamada caché, las configuraciones que esta puede tener y como esto influye en la rapidez del computador. [1]

En este laboratorio se solicita implementar 2 algoritmos de ordenamiento, uno iterativo y otro recursivo a elección. Estos deben ser implementados en lenguaje ensamblador para el procesador MIPS utilizando el simulador MARS, cada algoritmo constituye un programa y cada programa debe ser capaz de abrir un archivo de texto que contiene un número por línea, pasar estos números a memoria, ordenarlos y finalmente crear un nuevo archivo de texto con los números ordenados. Utilizando la herramienta Data Cache Simulator de MARS se deben probar las distintas configuraciones de caché y analizar los resultados.

Los algoritmos a implementar serán el Quicksort como recursivo y el Bubble Sort como iterativo. Para la implementación de estos algoritmos se utilizó el stack de MIPS para ordenar allí los números. Utilizando la herramienta Data Cache Simulator se obtienen los resultados que serán analizados en este informe.

Con la realización de este trabajo se busca comprobar empíricamente la materia vista en clases respecto a las configuraciones del cache, además de mejorar nuestras habilidades para programar en un lenguaje de muy bajo nivel como lo es el del procesador MIPS.

En el capítulo 2 se describen las configuraciones que puede tener el caché así como también conceptos claves en el entendimiento del cache y los algoritmos de ordenamiento. En el capítulo 3 se explica como fue llevado a cabo el desarrollo del trabajo. En el capítulo 4 se discuten los resultados obtenidos y se explican de acuerdo a la materia vista en clases y a investigación personal

CAPÍTULO 2. MARCO TEÓRICO

El pipeline es una técnica de implementación en la cual se traslapan las instrucciones durante la ejecución.

2.1 ETAPAS DEL PIPELINE

En MIPS se tienen 5 etapas: IF: En esta etapa se busca la instrucción en memoria. ID: Se leen los registros

mientras se decodifica la instrucción. EX: Se ejecuta la operación de la instrucción o bien se calcula una

dirección de memoria. MEM: En esta etapa se accede a un operando que se encuentra en memoria. WB:

Finalmente se escriben los resultados en un registro.

2.2 HAZARDS

Los hazards son obstáculos que se producen durante la ejecución del pipeline cuando una instrucción no se puede ejecutar en el siguiente ciclo de reloj. Existen 3 tipos de hazards:

CAPÍTULO 3. DESARROLLO

CAPÍTULO 4. DISCUSIONES DE LOS RESULTADOS

CAPÍTULO 5. CONCLUSIÓN

CAPÍTULO 6. BIBLIOGRAFÍA

- [1] Felipe Garay. «Laboratorio 3 Organización de Computadores: Pipeline». En: (2015).