

66:20 Organización de Computadoras

Trabajo práctico 3: Memorias caché

1. Objetivos

Familiarizarse con la medición de comportamiento utilizando herramientas de profiling y con la estimación de parámetros de una memoria caché a partir de patrones de acceso y tasa de desaciertos al implementar una suite de benchmarking de memorias caché.

2. Alcance

Este trabajo práctico es de elaboración grupal, evaluación individual, y de carácter obligatorio para todos alumnos del curso.

3. Requisitos

El trabajo deberá ser entregado personalmente, en la fecha estipulada, con una carátula que contenga los datos completos de todos los integrantes.

Además, es necesario que el trabajo práctico incluya (entre otras cosas, ver sección ??), la presentación de los resultados obtenidos, explicando, cuando corresponda, con fundamentos reales, las causas o razones de cada resultado obtenido.

El informe deberá respetar el modelo de referencia que se encuentra en el grupo, y se valorarán aquellos escritos usando la herramienta $\text{T}_{\text{E}}\text{X}$ / $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$.

4. Recursos

Para la realización de este trabajo práctico, se requiere la herramienta de debugging y profiling Valgrind[1], que corre en ambientes Linux-x86. En particular, se utilizará el módulo Cachegrind[2] para simulación de memorias caché.

5. Software a evaluar

5.1. Objetivo

Determinar el tamaño de bloque, la cantidad de vías y el tamaño total de una memoria caché de datos. Esta puede ser la simulada al utilizar la herramienta

Cachegrind o la propia del procesador donde se ejecute el suite de benchmarking desarrollado. Este trabajo es de libre realización, pero se aconseja seguir el modelo que se propone en la siguiente sección. Se pueden utilizar los lenguajes C o C++.

5.2. Modelo propuesto

1. Realizar un módulo (herramienta parser) que parsee los datos del archivo de salida del módulo Cachegrind para que obtenga la tasa de miss de la memoria caché de datos.
2. Realizar un módulo que ejecute una implementación del modelo de lazo simple para obtener el tamaño del bloque de la memoria caché midiendo la tasa de miss que proporciona la herramienta parser desarrollada en el paso anterior.
3. Realizar un módulo que ejecute una implementación del modelo de lazo simple para obtener el tamaño total de la memoria caché bajo estudio midiendo la tasa de miss que proporciona la herramienta parser desarrollada en el primer paso.
4. Realizar un módulo que ejecute una implementación del modelo de lazo simple para obtener la cantidad de vías de la memoria caché bajo estudio midiendo la tasa de miss que proporciona la herramienta parser desarrollada en el primer paso.
5. Realizar un programa que ejecute los módulos anteriores en el orden debido, proporcionando los datos que sean necesarios para cada uno y provea en su salida la información de la memoria caché estudiada.

5.3. Parámetros del Programa Principal

Se deberán implementar las opciones:

- -v, -version
- -h, -help
- -D1=Tamaño, vías, bytes x línea

En caso de existir el parámetro D1, éste debe pasarse a cachegrind para que simule una cache de esas características. En caso de existir errores, éstos deben ser impresos por *stderr*.

5.4. Salida del programa principal

La salida del programa principal debe ser por *stdout* y tener el siguiente formato:

```
$ ./tp2
Memoria Caché de Datos Analizada:
Vías: X Vías
Tamaño Total: Y Bytes
Tamaño de Bloque: Z Bytes
```

```
$ ./tp2 -D1=32768, 4, 32
Memoria Caché de Datos Analizada:
Vías: 4 Vías
Tamaño Total: 32768 Bytes
Tamaño de Bloque: 32 Bytes
```

6. Informe

El informe deberá incluir:

- Descripción de los módulos desarrollados.
- Detalle del funcionamiento de cada módulo explicando cómo se calcula cada uno de los parámetros pedidos.
- Modo de uso de la suite de benchmarking.
- Las corridas de prueba, con los comentarios pertinentes;
- Este enunciado.
- Un CD conteniendo todo el material digital. Por favor no pegarle etiquetas.

7. Fechas

Fecha de entrega: Jueves 18 de Mayo.

Referencias

- [1] Valgrind, <http://valgrind.org>.
- [2] Cachegrind, <http://valgrind.org/docs/manual/cg-manual.html>