CSC 325 Lab 7

Cachegrind

Table of Contents

This is an individual lab which must be completed on the ARM64 (aarch64) VM.

1. Lab Goals

1. Understand the cache behavior of row-major vs. column-major memory accesses.
2. Practice using valgrind's cachegrind and cg_annotate tools.

2. Lab Overview

Because caches significantly influence program performance, most systems provide profiling tools to measure a program's use of the cache. One such tool is valgrind's *cachegrind* mode, which this lab uses to evaluate cache performance.

3. Handy References

- Read and understand textbook chapter 11.5, especially 11.5.2
- View the `valgrind` manual page from the command-line:

```
$ man valgrind
```

4. Lab Starting Point Code

4.1. Getting the Starting Point Code

Download the starter files from Canvas:

```
Makefile        lab7.c
```

4.2. Starting Point Code

The files included are:

- `Makefile`: builds the `lab7.o` object file and the `lab7` executable file
- `lab7.c`: a slightly modified version of the textbook code in Chapter 11.5.2.

5. Lab Requirements

1. Install the valgrind tools on your VM, if they aren't already installed.

   ```
   sudo apt update
   sudo apt install valgrind
   ```

2. Build the files you downloaded from Canvas.

   ```
   make
   ```

3. Run the cachegrind tool on algorithm version 1.

   ```
   valgrind --tool=cachegrind ./lab7 1000 1
   ```

4. Run cg_annotate on the algorithm version 1 `cachegrind output file.`

   ```
   cg_annotate cachegrind.out.xxxxx > version1.txt
   ```

   where xxxxx is the processID number from algorithm version 1.

5. Run the cachegrind tool on algorithm version 2.

   ```
   valgrind --tool=cachegrind ./lab7 1000 2
   ```

6. Run cg_annotate on the algorithm version 2 `cachegrind output file.`

   ```
   cg_annotate cachegrind.out.yyyyy > version2.txt
   ```

   where yyyyy is the processID number from algorithm version 2.

7. Discuss the version1.txt and version2.txt results. Focus on the values that significantly differ: list their values, discuss what those values measured, and discuss the algorithm insight each difference provided you.

8. For full credit, your discussion should be neatly formatted and thoughtful.

6. Tips

If you forgot how to work with the aarch64 VM, see

Canvas | Files | Pages| aarch64 Assembly Language

7. Submitting your Lab

7.1 What to submit

- Your thoughtful discussion of the version1.txt and version2.txt results. **Submit in PDF format.**

- The version1.txt and version2.txt files.
- A screenshot showing the output from:

```
make clean
make
valgrind --tool=cachegrind ./lab7 1000 2
```

Zip your complete Lab7 folder including the screenshot and submit it to Canvas.