Contents

1 General Overview

The program first reads in the flags from the command line, and verifies them for correctness. Then it creates an array of structs to track the statistics of each inputted tracefile. Next, it creates a hash table for use as the page table, allocate memory for virtual memory. Then it reads all the references from files by calling readRefsFromFiles. It flushes the TLB if necessary, and then performs the necessary additions/lookups by calling addToMemory. Once computation has been completed, it displays it and performs cleanup for termination of the program.

1.1 Pointer to Code

c. Look in fileIO.c, and tvm379.c:161

d. Look at linkedlist.c:123 and memory.c:49

e. LRU implementation linkedlist.c:169

f. FIFO implementation in addNewNode linkedlist.c:40

g. Counters sprinkled through addToMemory:22

h. Dynamically grow lists into fixed size(tvm379.c:161) array and remove lists from array(tvm379.c:184)

i. See 'How Simulator is Efficient and Scalable'

j. See Results

k. See Results

2 How Simulator is Efficient and Scalable

2.1 fileIO.c

1. In line 54 it reads in the entire (4 * Quantum) bytes, rather than reading in 4 bytes quantum times. This removed a significant bottle neck in our code

2.2 memory.c

1. traceFileTracker is a "poor man's hash table" indexed by PID, that allows statistics to be tracked for each tracefile in constant time

2. On line 122 frameBuffer is a hash table used to store pointers to the current nodes in virtualMemory. Giving us near constant time access to seek the nodes.

3. We are tracking the last PID and last page number to determine automatically if the page is already in the TLB, and therefore do not need to search the TLB linearly. These are checked for before any searching is done

2.3 In General

1. We tried to minimize iteration as much as possible by having all arrays index on PID or some uniquely identifiable variable to ensure constant time lookup of data

2. We wrote tests found in the /tests folder to ensure our modules were working as intended

3. We maintained separate .c and .h files for each major functionality that are highly independent to ensure that code was easily debuggable and optimized

4. TLB, all pageTables, and virtualMemory are all represented as doubly linked list. This means that we can dynamically allocate all storage and only use up as much memory as we need at any given time and no more.

3 Testing

3.1 Test Suite

- Run "make test" to run the following test suite:

  time ./tvm379 1024 128 g 100 800 f heapsort-trace.bin quicksort-trace.bin

  time ./tvm379 1024 128 g 100 800 l heapsort-trace.bin quicksort-trace.bin

  time ./tvm379 1024 128 g 100 80 f heapsort-trace.bin quicksort-trace.bin

  time ./tvm379 1024 128 g 100 80 l heapsort-trace.bin quicksort-trace.bin

  time ./tvm379 1024 128 g 100 800 f heapsort-trace.bin

  time ./tvm379 1024 128 g 100 800 l heapsort-trace.bin

  time ./tvm379 1024 128 g 100 80 f heapsort-trace.bin

  time ./tvm379 1024 128 g 100 80 l heapsort-trace.bin

  time ./tvm379 1024 128 g 100 800 f quicksort-trace.bin

  time ./tvm379 1024 128 g 100 800 l quicksort-trace.bin

  time ./tvm379 1024 128 g 100 80 f quicksort-trace.bin

  time ./tvm379 1024 128 g 100 80 l quicksort-trace.bin

  time ./tvm379 65536 256 g 100 1000000 l heapsort-trace.bin quicksort-trace.bin quicksort-trace.bin heapsort-trace.bin quicksort-trace.bin

  time ./tvm379 16 8 g 10 1 l heapsort-trace.bin quicksort-trace.bin quicksort-trace.bin heapsort-trace.bin quicksort-trace.bin

  time ./tvm379 65536 256 g 100 1000000 f heapsort-trace.bin quicksort-trace.bin quicksort-trace.bin heapsort-trace.bin quicksort-trace.bin

  time ./tvm379 16 8 g 10 1 f heapsort-trace.bin quicksort-trace.bin quicksort-trace.bin heapsort-trace.bin quicksort-trace.bin

  time ./tvm379 65536 256 p 100 1000000 l heapsort-trace.bin quicksort-trace.bin quicksort-trace.bin heapsort-trace.bin quicksort-trace.bin

  time ./tvm379 16 8 p 10 1 l heapsort-trace.bin quicksort-trace.bin quicksort-trace.bin heapsort-trace.bin quicksort-trace.bin

  time ./tvm379 65536 256 p 100 1000000 f heapsort-trace.bin quicksort-trace.bin quicksort-trace.bin heapsort-trace.bin quicksort-trace.bin

  time ./tvm379 16 8 p 10 1 f heapsort-trace.bin quicksort-trace.bin quicksort-trace.bin heapsort-trace.bin quicksort-trace.bin

4 Results

* Test Machine Specifications

- All tests were run on a 2015 MBP
- Processor: 2.7Gh Intel Core i5
- Memory 8Gb 1867 Mhz DDR3

* Sample Output

```
time ./tvm379 1024 128 g 100 800 f heapsort-trace.bin quicksort-trace.bin
39749506 93234 92583 711
41154368 6872 6723 84
        6.86 real         6.70 user         0.12 sys
time ./tvm379 1024 128 g 100 800 l heapsort-trace.bin quicksort-trace.bin
39751281 56394 55723 718
41154496 7205 7076 77
       10.05 real         9.89 user         0.13 sys
time ./tvm379 1024 128 g 100 80 f heapsort-trace.bin quicksort-trace.bin
39116916 1978384 1978314 73
41108815 59985 59975 6
        5.14 real         4.99 user         0.11 sys
time ./tvm379 1024 128 g 100 80 l heapsort-trace.bin quicksort-trace.bin
39384876 1710424 1710354 73
41152198 16602 16592 6
        6.72 real         6.52 user         0.15 sys
time ./tvm379 1024 128 g 100 800 f heapsort-trace.bin
39766277 58011 57211 792
        5.84 real         5.72 user         0.08 sys
time ./tvm379 1024 128 g 100 800 l heapsort-trace.bin
39767314 45821 45021 792
        8.00 real         7.92 user         0.06 sys
time ./tvm379 1024 128 g 100 80 f heapsort-trace.bin
39138845 1956455 1956375 79
        3.91 real         3.84 user         0.06 sys
time ./tvm379 1024 128 g 100 80 l heapsort-trace.bin
39407890 1687410 1687330 79
        4.53 real         4.45 user         0.06 sys
time ./tvm379 1024 128 g 100 800 f quicksort-trace.bin
41163536 1798 998 792
```

```
        1.05 real         1.00 user         0.04 sys
time ./tvm379 1024 128 g 100 800 l quicksort-trace.bin
41163535 1792 992 792
        2.04 real         1.96 user         0.06 sys
time ./tvm379 1024 128 g 100 80 f quicksort-trace.bin
41162682 6118 6038 79
        1.07 real         1.01 user         0.05 sys
time ./tvm379 1024 128 g 100 80 l quicksort-trace.bin
41162732 6068 5988 79
        2.01 real         1.93 user         0.05 sys
time ./tvm379 65536 256 g 100 1000000 l heapsort-trace.bin quicksort-trace.bin quicksor
        41095283 17 0 16
41168784 16 0 15
41168784 16 0 15
41095284 16 0 15
41168784 16 0 15
       10.31 real         9.96 user         0.25 sys
time ./tvm379 16 8 g 10 1 l heapsort-trace.bin quicksort-trace.bin quicksort-trace.bin
0       41095320 41095320 0
0 41168880 41168880 0
0 41168880 41168880 0
0 41095320 41095320 0
0 41168880 41168879 0
       85.95 real        85.09 user         0.48 sys
time ./tvm379 65536 256 g 100 1000000 f heapsort-trace.bin quicksort-trace.bin quicksor
41095283 17 0 16
41168784 16 0 15
41168784 16 0 15
41095284 16 0 15
41168784 16 0 15
currentSize 81
        4.93 real         4.71 user         0.20 sys
time ./tvm379 16 8 g 10 1 f heapsort-trace.bin quicksort-trace.bin quicksort-trace.bin
21846743 19248577 19248577 0
17247291 23921589 23921589 0
17247291 23921589 23921589 0
21846743 19248577 19248577 0
17247291 23921589 23921588 0
currentSize 1
       49.88 real        48.12 user         0.61 sys
```

```
time ./tvm379 65536 256 p 100 1000000 l heapsort-trace.bin quicksort-trace.bin quicksor
38923571 17 0 16
40511517 16 0 15
40511517 16 0 15
38923572 16 0 15
40511517 16 0 15
currentSize 81
       11.27 real        10.97 user         0.25 sys
time ./tvm379 16 8 p 10 1 l heapsort-trace.bin quicksort-trace.bin quicksort-trace.bin
0 41095320 41095320 0
0 41168880 41168880 0
0 41168880 41168880 0
0 41095320 41095320 0
0 41168880 41168879 0
currentSize 1
       91.16 real        90.21 user         0.54 sys
time ./tvm379 65536 256 p 100 1000000 f heapsort-trace.bin quicksort-trace.bin quicksor
38923572 17 0 16
40511517 16 0 15
40511517 16 0 15
38923572 16 0 15
40511517 16 0 15
currentSize 81
        6.10 real         5.87 user         0.21 sys
time ./tvm379 16 8 p 10 1 f heapsort-trace.bin quicksort-trace.bin quicksort-trace.bin
21846743 19248577 19248577 0
17247291 23921589 23921589 0
17247291 23921589 23921589 0
21846743 19248577 19248577 0
17247291 23921589 23921588 0
currentSize 1
       52.90 real        52.24 user         0.40 sys
```