

ECE571: Advanced Microprocessor Design – Homework 6
Fall 2019

Due: Thursday 17 October 2019, 3:15pm

Create a document that contains the data and answers described in the sections below. A .pdf or .txt file is preferred but I can accept MS Office or Libreoffice format if necessary.

1. Bzip2 prefetch behavior on the x86 Haswell-EP Machine

For this section, log into the Haswell machine just like in previous homeworks.

Run the bzip2 benchmark on the Haswell machine.

- (a) Measure (in one command) bzip using the following events: `l2_rqsts.all_demand_references` which is total L2 cache accesses, `l2_rqsts.demand_data_rd_miss` which is total demand L2 cache misses, and `l2_rqsts.all_pf` which is total prefetches into the L2 cache.

```
perf stat -e l2_rqsts.all_demand_references:u,\
l2_rqsts.demand_data_rd_miss:u,\
l2_rqsts.all_pf:u \
/opt/ece571/401.bzip2/bzip2 -k -f ./input.source
```

Note, if the program finishes instantly with an error message, be sure you have `input.source` in your current directory. You can recopy it with

```
cp /opt/ece571/401.bzip2/input.source .
```

Calculate the L2 cache miss rate from the first two results (misses/total), the third event is just informational, it in theory tracks total number of prefetches.

Note all 3 values, the rate, and total time.

2. Software Prefetching and bzip2 on Haswell-EP

- (a) Re-run the previous prefetch results on Haswell, but instead of running `bzip2` run `bzip2.swprefetch` which was compiled with `-fprefetch-loop-arrays` which enables sw prefetch instructions.

Record the miss rate and total time.

```
perf stat -e l2_rqsts.all_demand_references:u,\
l2_rqsts.demand_data_rd_miss:u,\
l2_rqsts.all_pf:u \
/opt/ece571/401.bzip2/bzip2.swprefetch -k -f ./input.source
```

3. quake_l prefetch behavior on the x86 Haswell-EP Machine

Run `quake_l`:

```
(a) perf stat -e l2_rqsts.all_demand_references:u,\
    l2_rqsts.demand_data_rd_miss:u,\
    l2_rqsts.all_pf:u \
/opt/ece571/quake_1.speccomp/quake_1 < \
/opt/ece571/quake_1.speccomp/inp.in
```

Calculate the L2 cache miss rate from the first two results, also note the total time.

4. quake_1 software prefetch behavior on the x86 Haswell-EP Machine

Run quake_1 with software prefetch enabled:

```
(a) perf stat -e l2_rqsts.all_demand_references:u,\
    l2_rqsts.demand_data_rd_miss:u,\
    l2_rqsts.all_pf:u \
/opt/ece571/quake_1.speccomp/quake_1.swprefetch < \
/opt/ece571/quake_1.speccomp/inp.in
```

Calculate the L2 cache miss rate from the first two results, also note the total time.

5. Hardware Prefetch Disabled

It is possible to disable hardware prefetch on modern Intel processors.

See:

<https://software.intel.com/en-us/articles/disclosure-of-hw-prefetcher-control-on-some-intel-processors> for details.

It requires root permissions, so I have done the measurements for you, with the results summarized in the three tables below.

L2 extended results, HW Prefetch Disabled, (on regular Haswell, Haswell-EP is similar

benchmark	L2-total	L2-miss	L2-prefetches	time
bzip2	294,105,194	127,429,015	174,122	3.28
bzip2.swprefetch	293,392,001	128,848,679	197,188	3.25
quake_1	27,182,307,882	18,938,356,292	8,988,780	137.5
quake_1.swprefetch	28,005,813,185	19,127,383,783	8,390,162	137.4

Full Cache Results with Hardware Prefetch Enabled

benchmark	L1-miss	L2-miss	L3-miss	runtime
bzip2	5%	%	0.001%	4s
bzip2.swprefetch	5%	%	0.001%	4s
quake_1	6%	%	15%	35s
quake_1.swprefetch	6%	%	15%	35s

Full Cache Results with Hardware Prefetch Disabled

benchmark	L1-miss	L2-miss	L3-miss	runtime
bzip2	5%	45%	0.001%	4s
bzip2.swprefetch	5%	45%	0.001%	4s
quake_1	6%	60%	95%	57s
quake_1.swprefetch	6%	60%	95%	56s

6. Short Answer Questions

- (a) Did enabling software prefetch help on bzip2? (i.e. compare the results in question 1 and question 2?)
- (b) Did enabling software prefetch help on equake_1? (i.e. compare the results in question 3 and question 4?)
- (c) How did turning off the prefetcher affect the bzip2 results (i.e. question 1 vs question 5?)
- (d) How did turning off the prefetcher affect the equake_1 results (i.e. question 3 vs question 5?)
- (e) Based on the results in question 5, what level of cache is the hardware prefetcher fetching into?
- (f) With the hardware prefetcher disabled, did enabling software prefetch help at all? (question 5)
- (g) Why do you think the software prefetch performance is so underwhelming?

7. Submitting your work.

- Create the document containing the data as well as answers to the questions asked.
- Please make sure your name appears in the document.
- e-mail the file to me by the homework deadline.