ECE 574 – Cluster Computing
Lecture 8

Vince Weaver

http://www.eece.maine.edu/~vweaver

vincent.weaver@maine.edu

24 September 2015

Announcements

- Homework #3 will be posted eventually

Prefetch Latency Update

- It turns out prefetch instructions can have high latency after all.
- If various structures full, it might stall
- Noticed on Linux before, prefetch can trigger TLB miss handler (slow) especially if prefetching NULL pointer
- On our machine, instructions correlate with LLC misses as well as branch misses

- Agner Fog documents ivybridge prefetch as being very slow. Doesn't say much about Haswell.
- Results on pi2 more normal, but 400MB of samples?

Pthread Programming

Useful links:

- `https://computing.llnl.gov/tutorials/pthreads/`
- `http://www.cs.cf.ac.uk/Dave/C/node31.html`

Simple Pthread Example

See `pthread_simple.c`

Simple Init Example

See `pthread_init.c`.

Simple Init Example – continued

Some timing results on 2 core (4 thread) ivybridge:

1	0.331
2	0.220
3	0.200
4	0.148
5	0.157
6	0.143
7	0.157
8	0.142
16	0.168
32	0.189
64	0.161
128	0.162
256	0.179
512	0.181
1024	0.269
2048	0.489
4096	0.988

Simple Join Example

How to have one thread wait for another to finish.

See `pthread_join.c`

Stack Example

How to see how much stack is available, and how to change it if not enough.

See `pthread_stack.c`

Mutex Example

See `pthread_mutex.c`

- Can create mutes two ways,
  - Statically, when declared

    ```
    pthread_mutex_t our_mutex = PTHREAD_MUTEX_INITIALIZER;
    ```

  - Dynamically with `pthread_mutex_init()` which allows setting mutex object attributes, attr.
- The mutex is initially unlocked.
- Can specify protocol, priority ceiling, and if it's shared/private.

- lock, unlock, trylock. Lock will spin until available, trylock is non-blocking.

Deadlock

When you have more than one lock, it is possible to end up nesting locks in ways that lockup a program with both threads getting stuck.

Thread 1	Thread 2
pthread_mutex_lock(&mutex1);	pthread_mutex_lock(&mutex2);
pthread_mutex_lock(&mutex2);	pthread_mutex_lock(&mutex1);

Condition Variable Example?

Maybe next time

PAPI Example

See `pthread_papi.c`

- Initialize with:
  `PAPI_library_init(PAPI_VER_CURRENT);`

- You can/should check all functions to see if return `PAPI_OK`

- If using pthreads need to do:
  `PAPI_thread_init(pthread_self);`

- Eventsets are just integers

  ```
  int eventset=PAPI_NULL;
  ```

- Gathered results are typically 64-bit integers

  ```
  long long values[1];
  ```

- Create an eventset:

  ```
  PAPI_create_eventset(&eventset);
  ```

- Add an event. Available events can be seen with the papi_avail and papi_native_avail commands.

-
  ```
  PAPI_add_named_event(eventset,"PAPI_TOT_INS");
  ```

- Before the code of interest do a
  `PAPI_start(eventset);`

- Afterward do a
  `PAPI_stop(eventset,values);`
  and you can print the value or save it for later.