CS 110 Discussion 8

Multithreading with POSIX Threads

Yanjie Song

School of Information Science and Technology

April 2, 2020

Table of Contents

Table of Contents

Introduction on Multithreading

Definition

In computer architecture, **multithreading** is the ability of a central processing unit (CPU) (or a single core in a multi-core processor) to provide multiple threads of execution concurrently, supported by the operating system.

Introduction on Multithreading

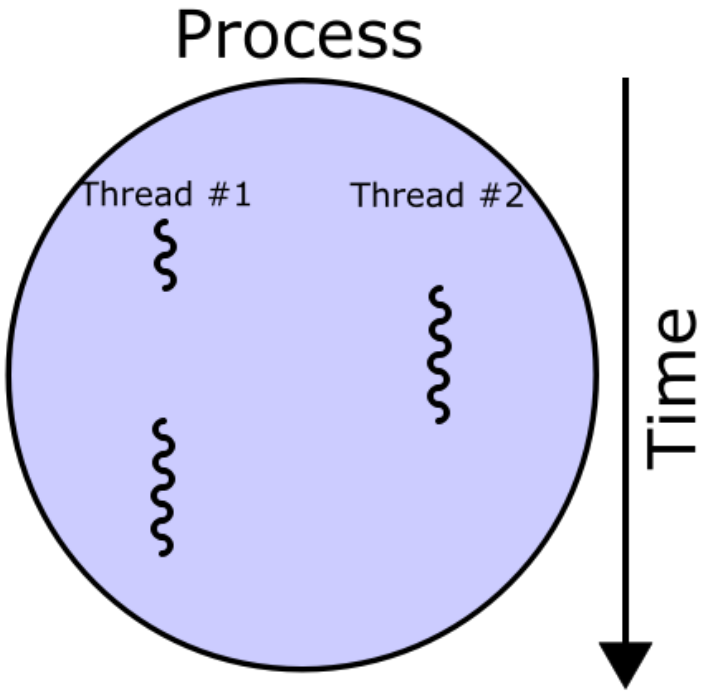

Figure: A process with two threads of execution, running on a single processor.

By I, Cburnett, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=2233446

Thread vs Process

Thread:

- The smallest execution unit
- Shared resources
- Small overhead

Process:

- The smallest allocation unit
- Self-managed resource
- Big overhead

Table of Contents

POSIX Threads

Definition

POSIX Threads, usually referred to as **pthreads**, is an execution model that exists independently from a language, as well as a parallel execution model. It allows a program to control multiple different flows of work that overlap in time. Each flow of work is referred to as a thread, and creation and control over these flows is achieved by making calls to the POSIX Threads API.

Pthreads in C

Most of the definitions are in the file `pthread.h`.
You need to pass specific arguments to the compiler to enable it.

- For example, pass `-pthread` to GCC during compilation.

Create a Thread

```
int pthread_create(pthread_t *thread, const pthread_attr_t
*attr, void *(*start_routine) (void *), void *arg)
```

Create a new thread

- `thread` will hold ID to new the thread after a successful call
- `attr` determines attributes for the new thread
- The new thread executes `start_routine` with `arg` as its parameter
- The thread function must have the following signature:
  `void *thread_func (void *arg)`
- Return 0 on success

Join a Thread

```
int pthread_join(pthread_t thread, void **retval);
```

Wait for a thread to terminate

- `thread` is the ID of the thread to wait for
- If `retval` is not `NULL`, it will hold the return value of the terminated thread
- Return 0 on success

Mutex Lock

Pthreads provides mutex lock for synchronization.

- `pthread_mutex_init`: Initailize a mutex lock
- `pthread_mutex_destroy`: Destory a mutex lock
- `pthread_mutex_lock`: Acquire the mutex lock
- `pthread_mutex_unlock`: Release the mutex lock

Other Synchronization Tools

Pthreads provides other tools for synchronization.

- `pthread_cond_t`: Condition variable
- `pthread_barrier_t`: Barrier

Table of Contents

Man Page

Definition

A **man page** (short for **manual page**) is a form of software documentation usually found on a Unix or Unix-like operating system.

Usage is very simple:

```
man [<man section>] <entry to lookup>
```

Manual Sections

On Linux, man pages often have 8 sections

1. General commands
2. System calls
3. Library functions, covering in particular the C standard library
4. Special files (usually devices, those found in /dev) and drivers
5. File formats and conventions
6. Games and screensavers
7. Miscellanea
8. System administration commands and daemons

There are also some subsections:

- p: POSIX specifications
- x: X Window System documentation

Thank you!

Demo code is on GitLab