CISC 3320
C12b: Thread Libraries

Hui Chen

Department of Computer & Information Science

CUNY Brooklyn College

Acknowledgement

- These slides are a revision of the slides provided by the authors of the textbook

Outline

- Thread Libraries
  - Pthread
  - Windows thread
  - Java thread and Java Executor Framework
- Implicit Threading
- Threading Issues
- Operating System Examples

Thread Libraries

- **Thread library** provides programmer with API for creating and managing threads
- Two primary ways of implementing
  - Library entirely in user space
  - Kernel-level library supported by the OS

Design Multithread Applications

- Two general strategies: asynchronous and synchronous threading
- Asynchronous threading
  - Parent thread creates child threads. Parent resumes after creation of the child threads
  - Parent and child threads execute concurrently and independently of one another
- Synchronous threading
  - Parent thread creates child threads. Parent then must wait for all of its children to terminate before it resumes.
  - Typically, synchronous threading involves significant data sharing among threads.
    - e.g., the parent thread may combine the results calculated by its various children.

Questions?

- Concept of thread libraries
- User and kernel space threads
- Asynchronous and synchronous threading

Pthreads

- May be provided either as user-level or kernel-level
- A POSIX standard (IEEE 1003.1c) API for thread creation and synchronization
- ***Specification***, not ***implementation***
- API specifies behavior of the thread library, implementation is up to development of the library
- Common in UNIX operating systems (Linux & Mac OS X)

Pthreads Example

- Essential APIs
  - pthread_attr_init
  - pthread_create
  - pthread_join
- How do threads share data? How do child thread inform the parent its exit status?
- The multithread π estimator application
- The multithread sum-all application

Questions?

- Using pthread
  - Essential APIs
  - How do threads share data?
- Are these two examples of synchronous threading or asynchronous threading?

Windows Threads

- Similar to Pthread from programing's (user's) perspective
- Essential APIs
  - CreateThread
  - WaitForSingleObject
  - WaitForMultipleObjects
    1. The number of objects to wait for
    2. A pointer to the array of objects
    3. A flag indicating whether all objects have been signaled
    4. A timeout duration (or INFINITE)

Windows Thread Example

- How do threads share data? How do child thread inform the parent its exit status?
- Windows Threads Example
  - The multithread sum-all application

Questions?

- Using windows threads
  - Essential APIs
  - How do threads share data?

Java Threads

- Threads are the fundamental model of program execution in a Java program.
  - The Java Virtual Machine (JVM) runs as a process
  - JVM runs a user program that consists of one or more threads

Creating Java Thread

- Essentially two methods
  1. Subclassing (extending) the Thread class
  2. Implementing the Runnable interface
- The 2nd approach is more commonly used
  - since a Java class can only extend one superclass, but can implement multiple interfaces and extend a superclass

Java Threads: Implementing Runnable

- Generally 3 steps
  - Create a class implementing the Runnable interface
  - Create a Thread object (passing an instance of the class in Step 1 as argument)
  - Start the thread

Java Threads: Implementing Runnable: Example

- The GUI multithread π estimator example

```
class Task implements Runnable {
  public void run() {
    // do the work
  }
}
Thread th = new Thread(new Task());
th.start();
```

Java Concurrent Package

- Since JDK 1.5, Java has introduced several new concurrency features
  - For much greater control over thread creation and communication.
- These tools are available in the java.util.concurrent package.

Java Executor Framework

- The Executor interface:

```
public interface Executor {
    void execute(Runnable command);
}
```

- Use it in this fashion

```
Executor executor = anExecutor;
executor.execute(new RunnableTask1());
executor.execute(new RunnableTask2());
```

JavaFX Concurrent Package

- For Java GUI application, JavaFX provides the javafx.concurrent package to work with JavaFX Task, Service, and ScheduledService

Questions?

- Concept of Java threads
- Creating Java threads, the essential method
- Java executor framework and others

Questions?

- Thread libraries
  - Pthreads
  - Windows threads
  - Java threads and Java Executor Framework