

Livelock Solutions

Livelock

- What is meant by the term livelock? How does it differ from deadlock?
 - A thread is livelocked when it is able to run, but cannot make any progress
 - In deadlock, the thread cannot run at all
- Briefly describe a situation where livelock can occur
 - Livelock can occur when trying to avoid deadlock
 - Instead of blocking indefinitely when they cannot get a lock, threads wait and retry
 - If the lock is not available, the threads will retry indefinitely
 - The threads are able to run, but cannot make any progress

Livelock

- Write a program which demonstrates livelock
- Suggest how livelock could be avoided in your program
 - Use `scoped_lock` to acquire both locks, or `unique_lock` with `lock()`
- Reimplement your program so it is not affected by livelock

Thread priority

- What is meant by thread priority? Is it supported in C++?
 - The priority of a thread is a number assigned to it by the operating system
 - A thread with high priority will be scheduled to run more often
 - A thread with low priority will sleep or be interrupted more often
 - C++ does not directly support thread priority
 - Thread priority can be set by calling an operating system API
 - The `native_handle` member function of the `std::thread` object will return the data which is needed for this call

Priority Inversion

- What is meant by priority inversion?
 - Priority inversion occurs when a high priority thread has to wait for a low priority thread
 - The low priority thread will be interrupted more often than the high priority thread
 - The high priority thread cannot do anything else while it is waiting for the low priority thread
 - In effect, the high priority thread runs at the same speed as a low priority thread

Resource starvation

- What is meant by resource starvation?
 - Resource starvation occurs when a thread cannot get the resources it needs to run
- Give some examples
 - Deadlock and livelock, in which the thread cannot obtain a lock it needs
 - A thread which cannot start because of insufficient operating system resources
 - In a badly designed scheduler, a low priority thread which does not run because there is always a higher priority thread