COMP 8551 Advanced Games Programming Techniques

Assignment Project Exam Help

https://powcoder.com

Add WeChat powcoder

Borna Noureddin, Ph.D.

British Columbia Institute of Technology

Realtime Issues and Multithreading II

Review

- Overview of multithreading
- Basic definitions
- Multithreading challenges
- Race conditions
- Mutexes

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

© Borna Noureddin

Overview

- Semaphores

Assignment Project Exam Help

https://powcoder.com

- Critical sections

Add WeChat powcoder

- Deadlocks

Semaphore

- Protected variable or abstract data type
- Synchronization method of controlling access by multiple processes to a common resource
- Binary semaphore (flag): true/false or locked/unlocked variable
- Counting semaphore: multiple access to shared resource

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

Semaphore

- Restaurant analogy:
  - Tables = “resources”
  - People = “threads” or “processes”
  - Host = “semaphore”
  - Host keeps track of unoccupied tables (utables) and who is to be seated next. Very focused and cannot be interrupted when performing duties.
  - Initially, utables = # of tables

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

Semaphore

- Restaurant analogy (cont'd):
  - When someone arrives, seated and utables updated as long as utables > 0
  - First come, first serve, but those with reservations may be seated ahead of others (priority)
  - If utables < 1, people wait in a queue for their table
  - When people leave, utables = utables + 1

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

© Borna Noureddin COMP 8551

Semaphore vs. Mutex

- Mutex = semaphore with only two values
- Mutex for single chair: only one person at a time can be sitting at it
- Semaphore for table with multiple chairs, or multiple tables in a restaurant: each table/chair can only be occupied by one group/person, but there are multiple tables/chairs
- Mutex more efficient than binary semaphore

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

© Borna Noureddin COMP 8551

Critical sections

General use of term (Wikipedia):

In concurrent programming a **critical section** is a piece of code that accesses a shared resource (data structure or device) that must not be concurrently accessed by more than one thread of execution. A critical section will usually terminate in fixed time, and a thread, task or process will have to wait a fixed time to enter it (aka bounded waiting).

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

© Borna Noureddin COMP 8551

Critical sections

- Kernel-level (vs. application-level):
  - Processes/threads cannot migrate to other processors
  - No pre-emption by other processes or interrupts
- Windows object:
  - More lightweight than mutex/semaphore/event
  - Can only be used within single process
  - See http://msdn.microsoft.com/en-us/library/ms682530(VS.85).aspx

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

Deadlocks

- One or more threads wait for resources that can never become available

Assignment Project Exam Help

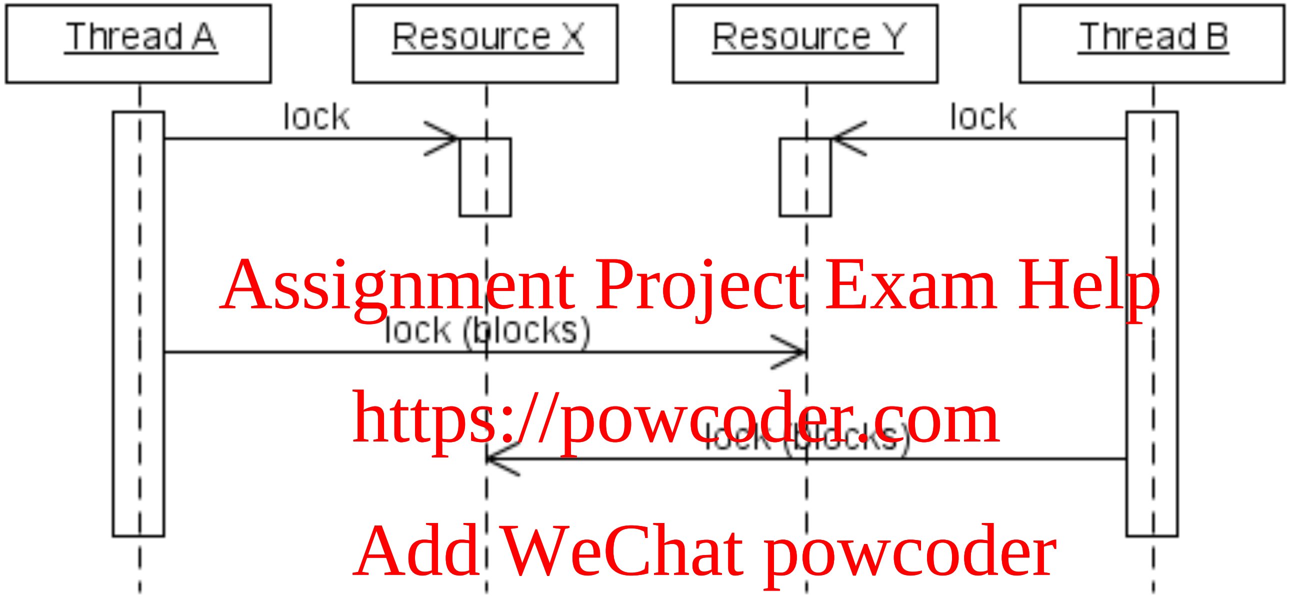
https://powcoder.com

- Classic case: two threads both require two shared resources, and use blocking mutexes to lock them in opposite order

Add WeChat powcoder

©-2012 Borna Noureddin

Deadlocks



Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

Thread A locks resource X

Thread B locks resource Y

Thread A attempts to lock resource Y

Thread B attempts to lock resource X

Both resources already locked (by the other thread): both threads wait indefinitely!

©-2012 Borna Noureddin

Deadlocks

- Necessary conditions:
  1. Mutual exclusion: a resource that cannot be shared by more than one process
  2. Hold and wait condition: processes already holding resources may request new resources
  3. No preemption condition: only a process holding a resource may release it
  4. Circular wait condition: two or more processes form a circular chain where each process waits for a resource that the next process in the chain holds

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

©-2012 Borna Noureddin COMP 8551

Deadlocks

Kansas legislature:

Assignment Project Exam Help

When two trains approach each other at a crossing, both shall come to a full stop and neither shall start up again until the other has gone.

https://powcoder.com

Add WeChat powcoder

©-2012 Borna Noureddin COMP 8551

Deadlock avoidance

- Check for availability before granting resource:
  - Will system enter an unsafe state?
  - System must know in advance number and type of all resources
  - E.g., Banker's algorithm
  - Normally, impossible to know in advance what every process will request

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

©-2012 Borna Noureddin COMP 8551

Deadlock avoidance

- Symmetry-breaking techniques:
  - Wait/Die and Wound/Wait
  - Process age determined by time stamp

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

	Wait/Die	**Wound/Wait**
O needs a resource held by Y	O waits	Y dies
Y needs a resource held by O	Y dies	Y waits

©-2012 Borna Noureddin COMP 8551

Deadlock prevention

- Remove mutual exclusion condition
  - Non-blocking synchronization algorithms
  - No exclusive access to resource
  - Impossible without spooling
  - Not foolproof even with spooling

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

Deadlock prevention

- Remove "hold and wait" conditions
  - Each process/thread must request all resources all at once (usually at startup)
  - Very difficult and inefficient
  - Alternative: release before request (all-or-none algorithms – not always practical)

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

©-2012 Borna Noureddin COMP 8551

Deadlock prevention

- Use timeouts
  - Only allowed to have resource for limited time
  - Difficult to reinforce
- Avoid circular wait condition
  - E.g., disable interrupts during critical sections
  - E.g., use a hierarchy to determine a partial ordering of resources

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

©-2012 Borna Noureddin COMP 8551

Deadlock detection

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

- OS or resource scheduler can detect deadlocks
- Roll back or restart one or more threads/processes
- Not always possible, and never guaranteed
- Generally, impossible to know if waiting for "unlikely" or "impossible" set of circumstances

©-2012 Borna Noureddin COMP 8551

Additional Reading

http://en.wikipedia.org/wiki/Semaphore_(programming)

http://en.wikipedia.org/wiki/Critical_section

Assignment Project Exam Help

http://msdn.microsoft.com/en-us/library/ms682530(VS.85).aspx

https://powcoder.com

http://www.drdobbs.com/high-performance-computing/225400066

Add WeChat powcoder

Review

- Semaphores

Assignment Project Exam Help

https://powcoder.com

- Critical sections

Add WeChat powcoder

- Deadlocks

© Borna Noureddin

Assignment Project Exam Help

END

https://powcoder.com

Add WeChat powcoder