Mohd Shoaib Rayeen
8 April 2019

Getblk Simulation

Using Multithreading, Mutex Lock and Conditional Variable

1) Problem Statement: The idea was to simulate Getblk algorithm that handle the allocation of buffers to processes. The requirement was to be able to visualise all 5 scenarios of Getblk and clearly identify the working of Getblk in all of those scenarios.
2) Programming Language Used: C++ is used for implementation. It can be implemented in C too but for signal and wait, C++ has thread, conditional variable and mutex libraries that simplify the implementation.
3) AOS Concepts Implemented: The algorithm needs multiprocessing, signals and wait so the followings are used:
    1) Thread - For Multithreading
    2) Mutex Lock - For Synchronisation
    3) Conditional Variable - For Signals
4) Getblk Cases Handled:
    1) Requested buffer is removed from FreeList and given to respective Thread.
    2) First block from FreeList is removed and given to respective Hash Queue.
    3) Asynchronous write to Disk and then add it at front of FreeList.
    4) Another thread frees the buffer and adds it to FreeList.
    5) Thread waits until buffer is free and if Buffer is free, then signal is raised.
5) Contribution and Learning Experience:

    Contribution: Algorithm is implemented by myself.

    Learning: 1) Learnt about Multithreading and MultiProcessing
    2) Different ways to implement it - Client-Server, Multithreading, Shared-Memory etc.
    3) Learnt about Synchronisation among Threads
    4) Learnt about Signals and waits in C++

Supervisor

Sapna Varshney