Fixed Priority Scheduling

In this way of scheduling threads, we give each thread a fixed priority. Out of all the runnable threads, the scheduler chooses the thread with the highest priority. When this thread terminates or waits, the scheduler picks the next thread with highest priority.

An example of keeping the list of threads could be in a priority queue, but in OS it is kept much simpler. If the thread with the highest priority has an integer from 1 to 10 where 1 is the thread with the highest priority, the way OS keeping track of it, is by having an array list where the indexes is a list of the threads with the priority number is equal the index number.

The scheduler handles ties in priority by either using round robin on the tied threads, or using first in first out on the first runnable thread.

It holds for fixed priority scheduling, that if threads meet their deadlines, then they will also do so under an assignment that prioritizes threads with shorter periods. Also the way to check that threads meet their deadlines, you check the worst-case situation, which is that all threads start at the same moment.