Andrew Tressler

Srilekha Dodda

CS3502

18 September 2025

Homework 1

Processes vs Threads

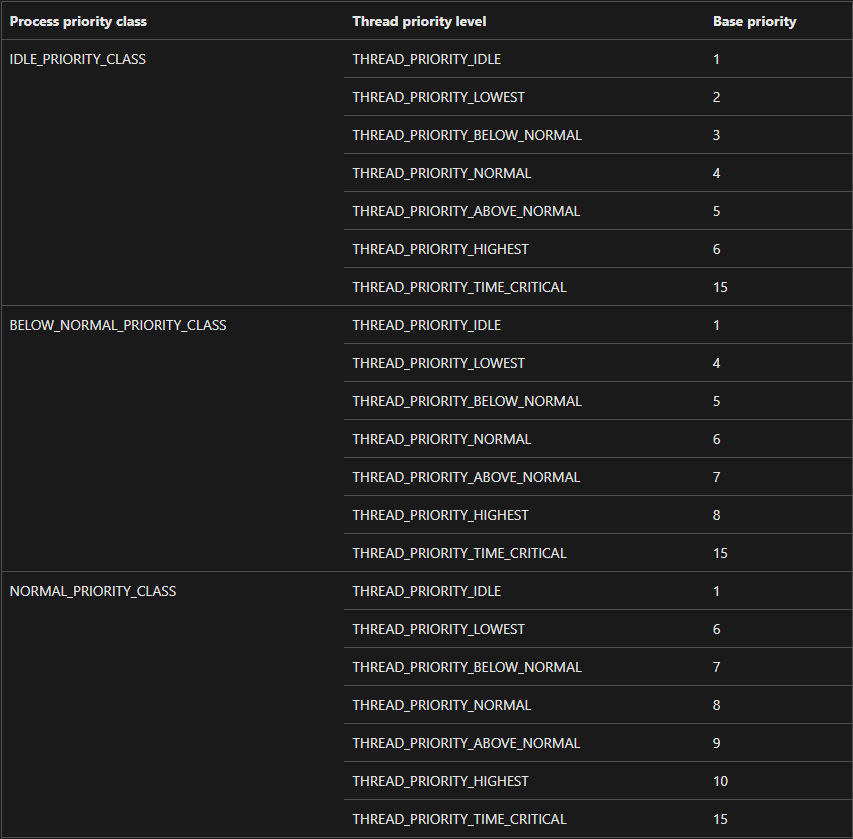
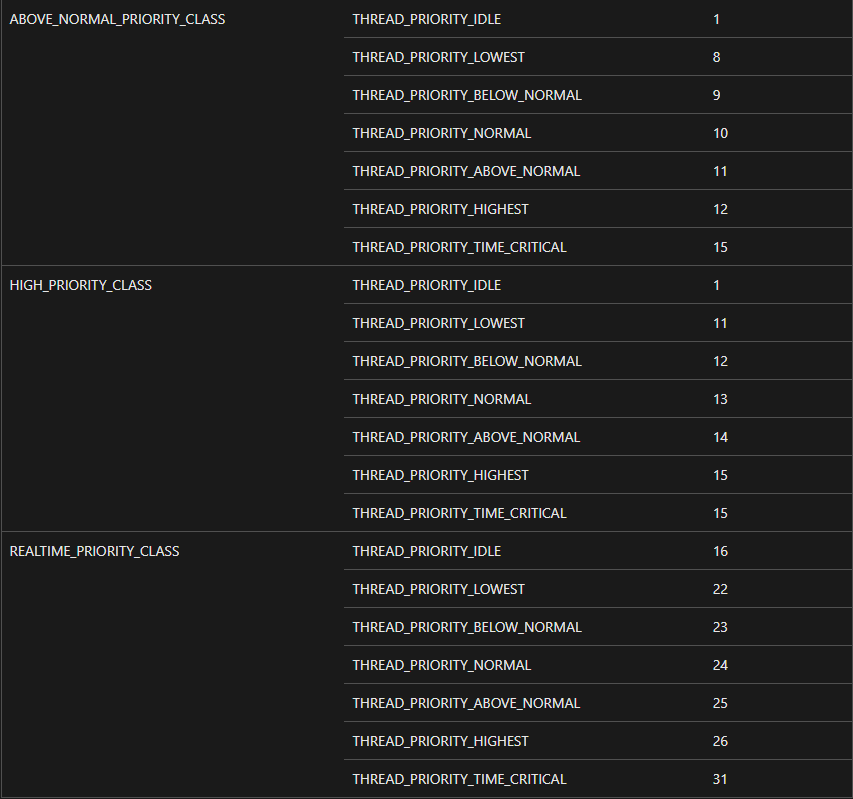
A process is defined as a program in execution. It is created when an executable (.exe) file is loaded into memory and has resources assigned for its execution. Each process must have at least one thread to execute, with the first one being called the primary thread.

Windows executes threads, not processes. Each process must have a thread to execute.

A process can create more threads to perform different tasks defined in the executable. Each thread of a given process will share it’s address space and sources. For example, one thread could handle user input, one could handle networking, one could handle rendering, etc. (1)

Process Scheduling

Windows uses round-robin style priority scheduling with preemption. This means that the highest priority threads will be given a piece of processing time (quantum) of size q. After q has elapsed, the scheduler will give q time to the next process in line (context switch). Threads with the highest priority will be executed first, and as they are completed, lower priority processes will start to be executed. If a low priority process is executing and a higher priority thread is made available, the lower priority thread will be preempted (halted) and a context switch will be performed to start executing the higher priority thread. Each process belongs to a certain process priority level, and each child thread of a given process is given a base priority number based on thread priority level. Below are the base priority values based on the Process priority class and thread priority level. (2)

The higher a thread’s priority is, the sooner it will get assigned compute time by the scheduler. Lower priority threads will have longer delays and may be starved for compute time if there are many high priority threads. To lessen the chance of visible delays to the user, Windows dynamically changes the priorities of processes that are either in the foreground or receiving user input using Priority Boosts. (3)

Process Context Switching

In Windows, a context switch is when a thread (including it’s registers and stacks) are stored to memory and a new thread is loaded in to the CPU and run. It is performed when:

* A higher priority thread becomes available for execution (preempts lower prio thread)
* The current thread’s time quantum (q) is over
* The current thread needs to wait

(4)

Works Cited

(1). “**Processes and Threads”**

<https://learn.microsoft.com/en-us/windows/win32/procthread/processes-and-threads>

(2). “**Scheduling Priorities”**

<https://learn.microsoft.com/en-us/windows/win32/procthread/scheduling-priorities>

(3). “**Priority Boosts”**

<https://learn.microsoft.com/en-us/windows/win32/procthread/priority-boosts>

(4). “**Context Switches**”

<https://learn.microsoft.com/en-us/windows/win32/procthread/context-switches>