Scheduler Comparison:

There are 3 schedulers in context – default scheduler, simple scheduler – where we implemented FIFO scheduler and advanced scheduler – where we implemented Priority scheduler.

The default scheduler of xv6 is Round Robin (mentioned in textbook). Round robin is a fair, preemptive scheduling algorithm that assigns CPU time to each process equally in a cyclic order, using a fixed time slice or quantum. It's particularly effective in time-sharing systems, ensuring that no single process monopolizes the CPU. RR improves responsiveness for a wide range of processes but may lead to overhead due to frequent context switches, especially if the time quantum is not optimally chosen.

First-In, First-Out (FIFO) schedules tasks in the exact order they arrive in the queue, operating on a non-preemptive basis. This straightforward method is easy to implement and understand, making it suitable for tasks with similar processing times. However, FIFO can lead to inefficient CPU utilization and longer average wait times, especially if longer tasks precede shorter ones, a problem known as the "convoy effect."

Priority Scheduling assigns a priority to each process, with the CPU being allocated to the process with the highest priority. It can be either preemptive or non-preemptive. This method allows more critical tasks to be addressed first, catering to the needs of real-time applications. However, without proper management, it can lead to starvation of lower-priority processes.

Simple Scheduler vs Default Scheduler:

Commands	Scheduler	ticks_running()	uptime()
ls	FIFO Scheduler	24	868
	Default	23	1147
find	FIFO Scheduler	114	950
	Default	113	1231
cat README|uniq	FIFO Scheduler	221	1054
	Default	221	1334
mkdir	FIFO Scheduler	328	1151
	Default	325	1428

Advanced Scheduler vs Default Scheduler:

Commands	Scheduler	ticks_running()	uptime()
ls	Priority Scheduler	25	1075
	Default	23	1147
find	Priority Scheduler	114	1159
	Default	114	1231
cat README|uniq	Priority Scheduler	218	1257
	Default	221	1334
mkdir	Priority Scheduler	307	1352
	Default	325	1428

Simple Scheduler vs Advanced Scheduler:

Commands	Scheduler	ticks_running()	uptime()
ls	FIFO Scheduler	24	868
	Priority Scheduler	25	1075
find	FIFO Scheduler	114	950
	Priority Scheduler	218	1159
cat README|uniq	FIFO Scheduler	221	1054
	Priority Scheduler	218	1857
mkdir	FIFO Scheduler	328	1151
	Priority Scheduler	307	1952

Discussion:

We have 3 schedulers – default which in xv6, is round robin scheduler, First In First Out (FIFO) scheduler and Priority Scheduler.

- - Default vs FIFO: The FIFO Scheduler tends to have lower turn-around times and higher efficiency compared to the Default Scheduler. This is because the processes scheduled have similar computational load and thus require a similar amount of processing time. If the test is run on longer processes, the default scheduler, which uses a round robin scheduler (mentioned in textbook) would fare better as FIFO can result in starvation.

- Default vs Priority: The Priority Scheduler significantly surpasses the Default Scheduler in efficiency and effectiveness. The average turnaround time for priority scheduler is lower than that of default scheduler. But some processes have much lower turnaround time than other processes. This is because processes with higher priority are immediately executed.

- FIFO vs Priority: There is no huge difference in performance between FIFO and Priority schedulers but average turnaround time for priority schedulers is higher than that of FIFO schedulers. The uptimes are also close but those of FIFO are smaller.