

Shahil Patel-200010039
Pranav Talegaonkar-200010041
Assignment-5
CS-314:Operating Systems Laboratory

1 Part-1

In this section, we had to set up *four* different workload mixtures, each with a different set of characteristics—from all-compute-intensive benchmarks to all-I/O to CPU-intensive workloads. Additionally, each task need to generate about 5 new processes.

It was necessary to modify the configuration files for the display of quanta before performing analysis on UnixBench. For that, a small modification was made to the ***system.c*** file located at ***minix/kernel/***. The quanta allotted and consumed were extracted in the manner described below in the function sched proc():

```
printf("Allotted Quantum is: %d, Used Quantum is: %d\n",  
p->p_quantum_size_ms,  
p->p_quantum_size_ms-cpu_time_2_ms(p->p_cpu_time_left));
```

This was sent to Minix3 for a build using the following script run1.sh:

```
cp system.c /usr/src/minix/kernel/;  
cd /usr/src/;  
make build MKUPDATE=yes
```

1.1 workload mix1.sh

In workload mix1.sh, the following script was added:

```
#!/bin/sh

./fstime.sh &

./fstime.sh &

./fstime.sh &

./fstime.sh &

./fstime.sh &

wait
```

In the above script, all the 5 processes were of fstime.sh with PIDs.

Observations & Inference: The nature of fstime.sh is inherently *I/O bound*. As a result, in this workload mix1.sh, each process would run its scripts in a *Round-Robin* fashion, consecutively. Five I/O-bound processes are scheduled to work on the CPU after waiting for their I/O task. According to the literature, I/O-bound processes don't use all of the allocated quanta (500). All processes wait for I/O to arrive before completing their tasks in that sequence.

```

2:59.45 real      3.83 user      28.25 sys
Minix: PID 466 exited
fstime completed
---
Minix: PID 458 exited
Copy done: 1000004 in 83.7167, score 2986
COUNT:2986:0:KBps
TIME:83.7
Minix: PID 461 exited
2:59.93 real      4.35 user      30.85 sys
Minix: PID 459 exited
fstime completed
---
Minix: PID 454 exited
Copy done: 1000004 in 83.8667, score 2980
COUNT:2980:0:KBps
TIME:83.9
Minix: PID 463 exited
3:00.05 real      4.23 user      30.83 sys
Minix: PID 460 exited
fstime completed
---
Minix: PID 455 exited
Minix: PID 453 exited
# _

```

workload__mix1.sh completed

1.2 workload mix2.sh

In workload mix2.sh, the following script was added:

```

#!/bin/sh

./syscall.sh &

./syscall.sh &

./syscall.sh &

./syscall.sh &

./syscall.sh &

wait

```

In the above script, all the 5 processes were of syscall.sh with PIDs.

Observations & Inference: These 5 processes run concurrently in Minix3's standard *Round-Robin* configuration. As previously indicated, this CPU-intensive operation has a 200 time limit. All of the quanta slots are not always fully engaged since *syscall.sh* is not as intensive as *arithoh.sh*. Round-robin scheduling is fairly implemented by CPU for all processes. We know from the literature that CPU-intensive processes don't require I/O and can complete their tasks in the time allotted to them. Thus, all processes in workload mix3.sh finish in nearly the same amount of time while only partially utilising the quantum slots that were allotted to them.

```
5:19.63 real      19.05 user      43.83 sys
Minix: PID 476 exited
syscall completed
---
Minix: PID 471 exited
Minix: PID 483 exited
5:20.56 real      19.46 user      44.75 sys
Minix: PID 480 exited
syscall completed
---
Minix: PID 473 exited
Minix: PID 477 exited
5:20.80 real      19.38 user      45.11 sys
Minix: PID 475 exited
syscall completed
---
Minix: PID 470 exited
Minix: PID 481 exited
5:21.08 real      19.63 user      45.61 sys
Minix: PID 478 exited
syscall completed
---
Minix: PID 472 exited
Minix: PID 469 exited
# _
```

workload__mix2.sh completed

1.3 workload mix3.sh

In workload mix3.sh, the following script was added:

```
#!/bin/sh

./pipe.sh &

./pipe.sh &

./syscall.sh &

./pipe.sh &

./pipe.sh &

wait
```

In the above script, 4 processes were of *pipe.sh* with PIDs and 1 processes were of *syscall.sh*.

Observations & Inference: syscall.sh is system CPU Intensive and pipe.sh is I/O Bound.

Clearly, CPU Intensive Tasks execute in a Round Robin fashion, whilst I/O bound processes wait for I/O for execution. Since, syscall.sh is a CPU Intensive Process which completely utilizes its allotted quanta and finishes its task and exits. Next, pipe.sh allotted more quanta (500) than a CPU Intensive task (200) and I/O Bound process doesn't utilize all quanta every time.

```
, Quantum Used is: 20
(200010039, 200010041)Minix: PID 33 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 20
(200010039, 200010041)Minix: PID 35 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 20
(200010039, 200010041)Minix: PID 255 swapped in
(200010039, 200010041) Quantum Alloted is: 500
, Quantum Used is: 500
Minix: PID 500 exited
      4:49.48 real      8.33 user      57.93 sys
Minix: PID 498 exited
pipe completed
---
Minix: PID 490 exited
(200010039, 200010041) Quantum Alloted is: 500
, Quantum Used is: 376
Minix: PID 496 exited
      4:51.05 real      8.01 user      58.80 sys
Minix: PID 493 exited
pipe completed
---
Minix: PID 488 exited
```

```
, Quantum Used is: 26
(200010039, 200010041)Minix: PID 255 swapped in
(200010039, 200010041) Quantum Alloted is: 500
, Quantum Used is: 500
Minix: PID 495 exited
      5:02.86 real      8.40 user      58.88 sys
Minix: PID 492 exited
pipe completed
---
Minix: PID 487 exited
Minix: PID 501 exited
      5:03.68 real      7.90 user      58.23 sys
Minix: PID 499 exited
pipe completed
---
Minix: PID 491 exited
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 200
(200010039, 200010041)Minix: PID 11 swapped in
(200010039, 200010041) Quantum Alloted is: 500
, Quantum Used is: 275
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 86
(200010039, 200010041)Minix: PID 11 swapped in
```

```

, Quantum Used is: 200
(200010039, 200010041)Minix: PID 11 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 200
(200010039, 200010041)Minix: PID 11 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 125
(200010039, 200010041)Minix: PID 11 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 200
(200010039, 200010041)Minix: PID 11 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 200
(200010039, 200010041)Minix: PID 11 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 141
(200010039, 200010041)Minix: PID 11 swapped in
Minix: PID 497 exited
    5:32.28 real    19.31 user      46.33 sys
Minix: PID 494 exited
syscall completed
---
Minix: PID 489 exited
Minix: PID 486 exited
# _

```

workload__mix3.sh completed

1.4 workload mix4.sh

In workload mix4.sh, the following script was added:

```

#!/bin/sh

./spawn.sh &

./spawn.sh &

./arithoh.sh &

./arithoh.sh &

./spawn.sh &

wait

```

In the above script, 3 processes were of *spawn.sh* with PIDs and 2 processes were of *arithoh.sh*.

Observations & Inference: These five processes run concurrently in Minix3's standard *Round-Robin* configuration. Such CPU-intensive jobs are given a time quanta by Minix3 that is 200 by default. Every time the process is scheduled during such executions, the 200 quanta allocation is given to both *spawn.sh*. We know from the literature that CPU-intensive processes don't require I/O and can complete their tasks in the time allotted to them. As a result, under this workload *mix4.sh*, *spawn.sh* will finish first as it is less CPU Intensive than *arithoh.sh*.

```
, Quantum Used is: 200
(200010039, 200010041)Minix: PID 51 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 115
(200010039, 200010041)Minix: PID 51 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 200
(200010039, 200010041)Minix: PID 51 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 200
(200010039, 200010041)Minix: PID 51 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 200
(200010039, 200010041)Minix: PID 51 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 200
(200010039, 200010041)Minix: PID 51 swapped in
Minix: PID 517 exited
      2:49.15 real      18.61 user      0.00 sys
Minix: PID 514 exited
arithoh completed
---
Minix: PID 509 exited
Minix: PID 506 exited
# _
```

workload__mix4.sh completed

2 Part-2

In this step, we had to adapt Minix3's user-level scheduler to the "Pseudo-FIFO" policy, which states that among all user-level processes that are prepared to run, the one that entered the system first must be scheduled. By making the following modifications, Minix3 was converted from its default behaviour of Round Robin Scheduling to Pseudo FIFO.

It was necessary to modify the configuration files for the display of quanta before performing analysis on UnixBench. To do that, a small modification was made to the file `schedule.c` located in `minix/servers/sched/`. The quanta allotted and consumed were extracted in the manner described below in the function `sched proc()`:

In the function `do_noquantum()`, following changes were made to prioritize in the queue:

```
rmp->priority -= 1; /* lower priority */
```

Now in order to balance the increase priority, we need to also make sure the queue length is not overflowed. For that, we can check the `balance_queues` function and change the following lines:

```
// rmp->priority -= 1; /* increase priority */
```

Basically, we want to comment the decrement here or else the priority queue will be overflowed.

2.1 workload mix1.sh

In workload mix1.sh, the following script was added:

```
#!/bin/sh

./fstime.sh &

./fstime.sh &

./fstime.sh &

./fstime.sh &

./fstime.sh &

wait
```

In the above script, all the 5 processes were of fstime.sh with PIDs.

Observations & Inference: The nature of fstime.sh is inherently I/O bound. The output makes it evident that the pseudo-FIFO order in these I/O-bound processes is not being followed correctly. This is the precise reason that this implementation is referred to as pseudo, and such circumstances may be exceptions that ultimately cause this procedure to be referred to as approximate or pseudo FIFO implementation.

Since I/O-bound processes are scheduled to run on the CPU when I/O is received, they are sent to the waiting queue after requesting I/O and then placed back in the ready queue. I/O-bound operations frequently do not use the entire 500-quanta slot given to them. They then proceed according to the standard Round-Robin Order. When Pseudo FIFO is used in this situation rather than Minix3's normal Round-Robin, there is no change seen.

Finally, we could see that, following our necessary code changes, CPU-intensive processes could adhere to correct FIFO scheduling whereas I/O-bound processes couldn't. This approximation and the exceptions cause us to categorise this scheduling as pseudo FIFO.

```

(200010039, 200010041)Minix: PID 23 swapped in
(200010039, 200010041) Quantum Alloted is: 500
, Quantum Used is: 110
(200010039, 200010041)Minix: PID 23 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 0
(200010039, 200010041)Minix: PID 25 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 5
(200010039, 200010041)Minix: PID 27 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 2
(200010039, 200010041)Minix: PID 28 swapped in
Read done: 1000004 in 9.8500, score 25380
COUNT:25380:0:KBps
TIME:9.8
Write done: 1008000 in 83.0667, score 3033
COUNT:3033:0:KBps
TIME:83.1
      1:27.11 real      5.20 user      28.01 sys
Minix: PID 380 exited
fstime completed
---
Minix: PID 375 exited

```

```

(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 200
Copy done: 1000004 in 16.5667, score 15090
COUNT:15090:0:KBps
TIME:16.6
Minix: PID 385 exited
      2:55.05 real      5.36 user      27.81 sys
Minix: PID 379 exited
fstime completed
---
Minix: PID 374 exited
Minix: PID 372 exited
# (200010039, 200010041) Quantum Alloted is: 500
, Quantum Used is: 465
(200010039, 200010041)Minix: PID 23 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 3
(200010039, 200010041)Minix: PID 25 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 11
(200010039, 200010041)Minix: PID 27 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 4
(200010039, 200010041)Minix: PID 28 swapped in

```

workload__mix1.sh completed

2.2 workload mix2.sh

In workload mix2.sh, the following script was added:

```
#!/bin/sh

./syscall.sh &

./syscall.sh &

./syscall.sh &

./syscall.sh &

./syscall.sh &

wait
```

In the above script, all the 5 processes were of syscall.sh with PIDs.

Observations & Inference: We may prioritize the 5 processes because they are all CPU-heavy but not as intensive as arithoh.sh. Here, the 5 CPU-intensive processes execute in a first-come, first-served order. Processes fully utilize their 200 quantum slots during such execution because they do not wait for I/O. Only when a process has finished running can the next one be scheduled. This difference is seen in Minix3 when Pseudo FIFO is used instead of the Round-Robin default.

```
, Quantum Used is: 200
(200010039, 200010041)Minix: PID 160 swapped in
Minix: PID 415 exited
      2:02.95 real      27.40 user      31.16 sys
Minix: PID 413 exited
syscall completed
---
```

```
Minix: PID 405 exited
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 200
(200010039, 200010041) Quantum Alloted is: 500
, Quantum Used is: 2
(200010039, 200010041)Minix: PID 23 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 2
(200010039, 200010041)Minix: PID 25 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 5
(200010039, 200010041)Minix: PID 27 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 2
(200010039, 200010041)Minix: PID 28 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 200
```

```
(200010039, 200010041) Quantum Alloted is: 500
, Quantum Used is: 0
(200010039, 200010041)Minix: PID 23 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 2
(200010039, 200010041)Minix: PID 25 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 6
(200010039, 200010041)Minix: PID 27 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 2
(200010039, 200010041)Minix: PID 28 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 147
(200010039, 200010041)Minix: PID 157 swapped in
Minix: PID 412 exited
      3:00.03 real      27.40 user      31.75 sys
Minix: PID 408 exited
syscall completed
---
```

```
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 200
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 200
```

workload__mix2.sh completed

2.3 workload mix3.sh

In workload mix3.sh, the following script was added:

```
#!/bin/sh

./pipe.sh &

./pipe.sh &

./syscall.sh &

./pipe.sh &

./pipe.sh &

wait
```

In the above script, 4 processes were of *pipe.sh* with PIDs and 1 processes were of *syscall.sh*.

Observations & Inference: Syscall.sh is scheduled in the script above until its execution is complete since scheduling is done in a pseudo-FIFO manner. Syscall.sh is finished ahead of ftime.sh because I/O bound processes are redirected to the ready queue and scheduled to work on the CPU after being dispatched to the waiting queue to request I/O. As a result, syscall.sh is finished prior to ftime.sh. Later, when I/O is received, ftime.sh is finished.

However, this is because the I/O bound processes are sent to the waiting queue after requesting I/O and are then placed back in the ready queue and scheduled to run on the CPU. In this execution, pipe.sh came before syscall.sh therefore in FIFO sequence, former had to be completed before later.

```
, Quantum Used is: 2
(200010039, 200010041)Minix: PID 25 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 7
(200010039, 200010041)Minix: PID 27 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 3
(200010039, 200010041)Minix: PID 28 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 2
(200010039, 200010041)Minix: PID 91 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 151
(200010039, 200010041)Minix: PID 179 swapped in
Minix: PID 449 exited
      1:18.31 real      28.15 user      30.68 sys
Minix: PID 445 exited
syscall completed
---
Minix: PID 440 exited
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 200
(200010039, 200010041) Quantum Alloted is: 500
, Quantum Used is: 500
```

```
, Quantum Used is: 0
(200010039, 200010041)Minix: PID 23 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 2
(200010039, 200010041)Minix: PID 25 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 6
(200010039, 200010041)Minix: PID 27 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 3
(200010039, 200010041)Minix: PID 28 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 1
(200010039, 200010041)Minix: PID 91 swapped in
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 195
(200010039, 200010041)Minix: PID 181 swapped in
Minix: PID 451 exited
      5:23.38 real     11.91 user     54.35 sys
Minix: PID 448 exited
pipe completed
---
Minix: PID 441 exited
Minix: PID 437 exited
# _
```

workload__mix3.sh completed

2.4 workload mix4.sh

In workload mix4.sh, the following script was added:

```
#!/bin/sh

./spawn.sh &

./spawn.sh &

./arithoh.sh &

./arithoh.sh &

./spawn.sh &

wait
```

In the above script, 3 processes were of *spawn.sh* with PIDs and 2 processes were of *arithoh.sh*.

Observations & Inference: In the above script, *arithoh.sh* is repetitively scheduled till its execution is done as the scheduling is in Pseudo FIFO fashion. Both *arithoh.sh* are consecutively scheduled after that *spawn.sh* is scheduled. In the next timeline, *spawn.sh* is completed.


```
Minix: PID 764 exited
Minix: PID 697 exited
Minix: PID 765 created
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 200
(200010039, 200010041)Minix: PID 117 swapped in
10.68Minix: PID 765 exited
sys
Minix: PID 703 exited
Minix: PID 766 created
(200010039, 200010041) Quantum Alloted is: 200
, Quantum Used is: 200
(200010039, 200010041)Minix: PID 118 swapped in
spawn completed
---
Minix: PID 766 exited
Minix: PID 698 exited
Minix: PID 719 exited
3:00.11 real    0.40 user    16.26 sys
Minix: PID 710 exited
spawn completed
---
Minix: PID 701 exited
Minix: PID 696 exited
#
```

workload__mix4.sh completed