

# CS 503 Fall 2022: Lab 3

Suraj Aralihalli

October 2022

## 3. Monitoring process resource consumption, performance, and behavior

### 3.2 Frequency of CPU usage

Update to currcount is made in:

- create.c - initialized to 0 when a new process is created
- initialize.c - initialized to 0 for null process
- resched.c - incremented when a process context switches in

### 3.4 Preemption types

My approach to distinguish between 2 types of preemption is by using a variable called `preemptionType` (defined with `extern` in `process.h`). I have used the following convention:

1. `preemptionType = 0` indicates No preemption
2. `preemptionType = 1` indicates Type 1 preemption
3. `preemptionType = 2` indicates Type 2 preemption

In `initialize.c` I set the `preemptionType` to 0 because when the OS first starts there is no preemption. In `clkhandler.c` I set the `preemptionType` to 1 in the if block when current process exhausts its quantum. Additionally, `preemptionType` is reset to 0 after returning from `resched()`.

```
if((--preempt) <= 0) {
    preemptionType = 1;
    preempt = QUANTUM;
    resched();
    preemptionType = 0;
}
```

Similarly, I set the `preemptionType` to 2 in the if block when someother process wakes up. Additionally, `preemptionType` is reset to 0 after returning from `wakeup()`.

```

if(!isempty(sleepq)) {
    if((--queuetab[firstid(sleepq)].qkey) <= 0) {
        preemptionType = 2;
        wakeup();
        preemptionType = 0;
    }
}

```

Setting the flag in clkhandler to 1 or 2 does not always mean there was a type 1 or type 2 preemption. In the resched() function, and within the scope of `if (ptold->prstate == PR\_CURR)` block when priority of next process in the queue is greater or equal to current process's priority, I check and update the preemption counts.

```

if(preemptionType == 1)
{
    ptold->prpreemptcount1++;
}
else if(preemptionType == 2)
{
    ptold->prpreemptcount2++;
}
preemptionType = 0;

```

Because, only at this point it is guaranteed that there was preemption (by a process with equal or higher priority) and the type of preemption can be captured.

## 4. Dynamic priority scheduling using multilevel feedback queue

### 4.5 Performance evaluation

**Benchmark A** Figure 1 shows the results when 8 cpubound processes are created using creatertx() without starvation prevention enabled. All the processes have similar values for all the metrics. Having identical max-response-time and average-response-time is an indicator that no process had to wait longer than the other for the cpu. They all have similar cpu-usage and user-cpu-usage because they are running the same operations i.e func cpubound(). preemption2-count is 0 because none of these processes go through a sleep()/wakeup() regime. These processes also context switch equal number of times and have similar preemption1-count. Figure 8 shows the results when starvation prevention is enabled.

**Benchmark B** Figure 2 shows the results when 8 iobound processes are created using creatertx() without starvation prevention enabled. They all have similar cpu-usage and user-cpu-usage because they are running the same operations i.e func iobound(). However, these values are lower compared to cpubound usage metrics even though both the benchmarks were run for 10,000 milli seconds. This is because major chunk of time spent by iobound workload is in sleep(). When the process goes to sleep(), the process state is changed to PR\\_SLEEP until it is put back to the multilevel feedback queue. Consequently, sleep() doesnot contribute to usercpu or totalcpu time. The max-response-time and average-response-time for the iobound workloads is similar and steeply lesser than the response times for cpubound workloads. This happens because, when an iobound process wakes up and process state changes to PR\\_READY, it is immediately awarded the cpu (as priority is

```
#####
Greetings from Suraj Aralihalli!
Username: saraliha
#####

cpubound, pid:7, cpu-usage:1230, user-cpu-usage:1286, average-response-time:637, max-response-time:718, number-context-switches:14, preemption1-count:13, preemption2-count:0
cpubound, pid:8, cpu-usage:1230, user-cpu-usage:1200, average-response-time:644, max-response-time:719, number-context-switches:14, preemption1-count:13, preemption2-count:0
cpubound, pid:9, cpu-usage:1230, user-cpu-usage:1200, average-response-time:645, max-response-time:718, number-context-switches:14, preemption1-count:13, preemption2-count:0
cpubound, pid:10, cpu-usage:1230, user-cpu-usage:1200, average-response-time:647, max-response-time:718, number-context-switches:14, preemption1-count:13, preemption2-count:0
cpubound, pid:11, cpu-usage:1230, user-cpu-usage:1200, average-response-time:648, max-response-time:718, number-context-switches:14, preemption1-count:13, preemption2-count:0
cpubound, pid:2, cpu-usage:1333, user-cpu-usage:1300, average-response-time:599, max-response-time:718, number-context-switches:15, preemption1-count:14, preemption2-count:0
cpubound, pid:5, cpu-usage:1333, user-cpu-usage:1300, average-response-time:600, max-response-time:718, number-context-switches:15, preemption1-count:14, preemption2-count:0
cpubound, pid:6, cpu-usage:1333, user-cpu-usage:1300, average-response-time:601, max-response-time:718, number-context-switches:15, preemption1-count:14, preemption2-count:0
```

Figure 1: Benchmark A without starvation prevention enabled

boosted). iobound processes execute sleep() inside a while loop. Consequently, iobound processes are context switched out/in more often and have higher currcount value compared to cpubound processes. This also explains why iobound processes have lower preemption1 count compared to cpu bound processes (they voluntarily relinquish the cpu). The difference in preemption2 count in iobound workload is due to the order in which these processes were created (Note that main()'s priority is decremented when it is still creating the workload processes). Figure 9 shows the results when starvation prevention is enabled.

```
#####
Greetings from Suraj Aralihalli!
Username: saraliha
#####

iobound, pid:5, cpu-usage:192, user-cpu-usage:189, average-response-time:0, max-response-time:1, number-context-switches:193, preemption1-count:0, preemption2-count:70
iobound, pid:9, cpu-usage:194, user-cpu-usage:151, average-response-time:0, max-response-time:7, number-context-switches:151, preemption1-count:0, preemption2-count:27
iobound, pid:7, cpu-usage:194, user-cpu-usage:159, average-response-time:0, max-response-time:4, number-context-switches:160, preemption1-count:0, preemption2-count:36
iobound, pid:2, cpu-usage:195, user-cpu-usage:124, average-response-time:0, max-response-time:0, number-context-switches:125, preemption1-count:0, preemption2-count:0
iobound, pid:6, cpu-usage:195, user-cpu-usage:124, average-response-time:0, max-response-time:3, number-context-switches:125, preemption1-count:0, preemption2-count:0
iobound, pid:8, cpu-usage:195, user-cpu-usage:124, average-response-time:0, max-response-time:6, number-context-switches:125, preemption1-count:0, preemption2-count:0
iobound, pid:10, cpu-usage:195, user-cpu-usage:124, average-response-time:0, max-response-time:9, number-context-switches:125, preemption1-count:0, preemption2-count:0
iobound, pid:11, cpu-usage:195, user-cpu-usage:125, average-response-time:0, max-response-time:11, number-context-switches:125, preemption1-count:0, preemption2-count:0
```

Figure 2: Benchmark B without starvation prevention enabled

**Benchmark C** Figure 3 shows the results when 4 iobound processes followed by 4 cpubound processes are created using creatertx() without starvation prevention enabled. Both usercpu and totalcpu time of cpubound workload is greater than iobound workload as expected. The preemption count of iobound processes is also lesser than cpu bound processes because priority of iobound processes is boosted every time the process undergoes sleep(). Additionally, the priority of cpu bound processes is decremented every time it consumes the quantum. Avg response time of cpubound workload is higher because they have the least priority in the feedback queue. Figure 10 shows the results when starvation prevention is enabled.

```
#####
Greetings from Suraj Aralihalli!
Username: saraliha
#####

cpubound, pid:11, cpu-usage:2355, user-cpu-usage:2373, average-response-time:53, max-response-time:183, number-context-switches:147, preemption1-count:24, preemption2-count:122
cpubound, pid:9, cpu-usage:2292, user-cpu-usage:2305, average-response-time:50, max-response-time:82, number-context-switches:157, preemption1-count:24, preemption2-count:132
cpubound, pid:8, cpu-usage:2402, user-cpu-usage:2404, average-response-time:57, max-response-time:281, number-context-switches:138, preemption1-count:25, preemption2-count:112
cpubound, pid:10, cpu-usage:2393, user-cpu-usage:2401, average-response-time:52, max-response-time:104, number-context-switches:149, preemption1-count:25, preemption2-count:123
iobound, pid:2, cpu-usage:195, user-cpu-usage:124, average-response-time:0, max-response-time:0, number-context-switches:125, preemption1-count:0, preemption2-count:0
iobound, pid:6, cpu-usage:195, user-cpu-usage:124, average-response-time:0, max-response-time:3, number-context-switches:125, preemption1-count:0, preemption2-count:0
iobound, pid:7, cpu-usage:195, user-cpu-usage:125, average-response-time:0, max-response-time:4, number-context-switches:125, preemption1-count:0, preemption2-count:0
iobound, pid:5, cpu-usage:195, user-cpu-usage:127, average-response-time:0, max-response-time:1, number-context-switches:127, preemption1-count:0, preemption2-count:2
```

Figure 3: Benchmark C without starvation prevention enabled

**Benchmark D** Figure 4 shows the results when 63 iobound processes along with 1 cpubound process are created using creatertx() without starvation prevention enabled. And Figure 5 shows the results for the same workload with starvation prevention enabled. The avg response time for the cpubound process when starvation prevention is disabled is 2628 ms and the max response time is 9512ms. However, when starvation prevention is enabled, the avg response time of the cpu bound process drastically drops to 106ms and max response time drops to 951ms. Starvation prevention has a minor influence on iobound process for avg response time. The avg of avg response time of iobound processes without starvation prevention is 2ms. The value slightly increases to 8ms when starvation prevention is enabled. The usercpu of cpu bound process with starvation prevention enabled is higher than when it is disabled indicating the cpu workload was given a better chance to run when starvation prevention is enabled. I have used the default values of STARVATIONTHRESHOLD, STARVATIONPERIOD, PRIOBOOST to carry out this experiment.

## 5. Bonus

Figure 7 and 6 show results when main is created using createetx with priority of 10 (RT process) and Benchmark A is run with cpubound process having initial priority of TSXINIT. In the previous experiments, the priority of main is decremented while running the creatertx() syscalls to generate

```

iobound, pid:44, cpu-usage:185, user-cpu-usage:184, average-response-time:2, max-response-time:63, number-context-switches:219, preemption1-count:0, preemption2-count:101
iobound, pid:8, cpu-usage:187, user-cpu-usage:180, average-response-time:2, max-response-time:6, number-context-switches:212, preemption1-count:0, preemption2-count:93
iobound, pid:31, cpu-usage:185, user-cpu-usage:184, average-response-time:2, max-response-time:42, number-context-switches:227, preemption1-count:0, preemption2-count:109
iobound, pid:40, cpu-usage:185, user-cpu-usage:187, average-response-time:2, max-response-time:57, number-context-switches:226, preemption1-count:0, preemption2-count:108
iobound, pid:36, cpu-usage:185, user-cpu-usage:181, average-response-time:2, max-response-time:50, number-context-switches:228, preemption1-count:0, preemption2-count:110
iobound, pid:46, cpu-usage:185, user-cpu-usage:188, average-response-time:2, max-response-time:66, number-context-switches:225, preemption1-count:0, preemption2-count:107
iobound, pid:55, cpu-usage:185, user-cpu-usage:191, average-response-time:2, max-response-time:80, number-context-switches:218, preemption1-count:0, preemption2-count:100
iobound, pid:42, cpu-usage:185, user-cpu-usage:185, average-response-time:2, max-response-time:60, number-context-switches:233, preemption1-count:0, preemption2-count:115
iobound, pid:62, cpu-usage:146, user-cpu-usage:165, average-response-time:13, max-response-time:313, number-context-switches:195, preemption1-count:0, preemption2-count:102
iobound, pid:6, cpu-usage:187, user-cpu-usage:186, average-response-time:2, max-response-time:5, number-context-switches:226, preemption1-count:0, preemption2-count:107
iobound, pid:18, cpu-usage:187, user-cpu-usage:182, average-response-time:2, max-response-time:22, number-context-switches:222, preemption1-count:0, preemption2-count:103
iobound, pid:5, cpu-usage:187, user-cpu-usage:181, average-response-time:2, max-response-time:4, number-context-switches:224, preemption1-count:0, preemption2-count:105
iobound, pid:21, cpu-usage:187, user-cpu-usage:172, average-response-time:2, max-response-time:27, number-context-switches:211, preemption1-count:0, preemption2-count:92
iobound, pid:41, cpu-usage:185, user-cpu-usage:189, average-response-time:2, max-response-time:58, number-context-switches:239, preemption1-count:0, preemption2-count:121
iobound, pid:30, cpu-usage:185, user-cpu-usage:201, average-response-time:2, max-response-time:41, number-context-switches:239, preemption1-count:0, preemption2-count:121
iobound, pid:12, cpu-usage:187, user-cpu-usage:176, average-response-time:2, max-response-time:12, number-context-switches:223, preemption1-count:0, preemption2-count:104
iobound, pid:10, cpu-usage:187, user-cpu-usage:181, average-response-time:2, max-response-time:9, number-context-switches:223, preemption1-count:0, preemption2-count:104
cpubound, pid:67, cpu-usage:0, user-cpu-usage:3, average-response-time:2628, max-response-time:9512, number-context-switches:4, preemption1-count:0, preemption2-count:3
iobound, pid:15, cpu-usage:187, user-cpu-usage:182, average-response-time:1, max-response-time:17, number-context-switches:233, preemption1-count:0, preemption2-count:114
iobound, pid:58, cpu-usage:179, user-cpu-usage:178, average-response-time:3, max-response-time:171, number-context-switches:229, preemption1-count:0, preemption2-count:115
iobound, pid:45, cpu-usage:185, user-cpu-usage:181, average-response-time:2, max-response-time:65, number-context-switches:237, preemption1-count:0, preemption2-count:119
iobound, pid:20, cpu-usage:187, user-cpu-usage:181, average-response-time:2, max-response-time:25, number-context-switches:221, preemption1-count:0, preemption2-count:102
iobound, pid:27, cpu-usage:187, user-cpu-usage:175, average-response-time:2, max-response-time:36, number-context-switches:218, preemption1-count:0, preemption2-count:99
iobound, pid:17, cpu-usage:187, user-cpu-usage:175, average-response-time:2, max-response-time:20, number-context-switches:221, preemption1-count:0, preemption2-count:102
iobound, pid:23, cpu-usage:187, user-cpu-usage:186, average-response-time:2, max-response-time:30, number-context-switches:221, preemption1-count:0, preemption2-count:102
iobound, pid:14, cpu-usage:187, user-cpu-usage:177, average-response-time:2, max-response-time:16, number-context-switches:224, preemption1-count:0, preemption2-count:105
iobound, pid:13, cpu-usage:187, user-cpu-usage:182, average-response-time:2, max-response-time:14, number-context-switches:229, preemption1-count:0, preemption2-count:110
iobound, pid:38, cpu-usage:187, user-cpu-usage:170, average-response-time:2, max-response-time:53, number-context-switches:215, preemption1-count:0, preemption2-count:96

```

Figure 4: Benchmark D without starvation prevention enabled

the workload processes and is context switched out by the child processes even before all the child processes are created and put in the multilevel feedback queue. With this change the main is now created as an RT process and retains its priority 10 through out its lifetime.

iobound, pid:28, cpu-usage:165, user-cpu-usage:173, average-response-time:7, max-response-time:38, number-context-switches:210, preemption1-count:0, preemption2-count:105  
iobound, pid:62, cpu-usage:133, user-cpu-usage:132, average-response-time:21, max-response-time:934, number-context-switches:156, preemption1-count:0, preemption2-count:71  
iobound, pid:8, cpu-usage:164, user-cpu-usage:172, average-response-time:8, max-response-time:49, number-context-switches:200, preemption1-count:0, preemption2-count:95  
iobound, pid:19, cpu-usage:165, user-cpu-usage:168, average-response-time:8, max-response-time:48, number-context-switches:198, preemption1-count:0, preemption2-count:93  
iobound, pid:59, cpu-usage:138, user-cpu-usage:136, average-response-time:18, max-response-time:749, number-context-switches:166, preemption1-count:0, preemption2-count:78  
iobound, pid:56, cpu-usage:149, user-cpu-usage:157, average-response-time:13, max-response-time:228, number-context-switches:190, preemption1-count:0, preemption2-count:95  
iobound, pid:57, cpu-usage:141, user-cpu-usage:127, average-response-time:17, max-response-time:650, number-context-switches:164, preemption1-count:0, preemption2-count:74  
iobound, pid:36, cpu-usage:164, user-cpu-usage:160, average-response-time:8, max-response-time:78, number-context-switches:193, preemption1-count:0, preemption2-count:88  
iobound, pid:23, cpu-usage:164, user-cpu-usage:151, average-response-time:8, max-response-time:63, number-context-switches:197, preemption1-count:0, preemption2-count:92  
iobound, pid:16, cpu-usage:166, user-cpu-usage:150, average-response-time:8, max-response-time:63, number-context-switches:182, preemption1-count:0, preemption2-count:76  
iobound, pid:45, cpu-usage:164, user-cpu-usage:162, average-response-time:8, max-response-time:77, number-context-switches:193, preemption1-count:0, preemption2-count:88  
iobound, pid:37, cpu-usage:165, user-cpu-usage:160, average-response-time:8, max-response-time:61, number-context-switches:196, preemption1-count:0, preemption2-count:91  
iobound, pid:27, cpu-usage:163, user-cpu-usage:157, average-response-time:8, max-response-time:77, number-context-switches:203, preemption1-count:0, preemption2-count:99  
iobound, pid:38, cpu-usage:164, user-cpu-usage:164, average-response-time:8, max-response-time:77, number-context-switches:193, preemption1-count:0, preemption2-count:88  
iobound, pid:64, cpu-usage:126, user-cpu-usage:130, average-response-time:21, max-response-time:927, number-context-switches:168, preemption1-count:0, preemption2-count:87  
iobound, pid:17, cpu-usage:163, user-cpu-usage:170, average-response-time:8, max-response-time:31, number-context-switches:210, preemption1-count:0, preemption2-count:106  
iobound, pid:31, cpu-usage:165, user-cpu-usage:158, average-response-time:8, max-response-time:42, number-context-switches:197, preemption1-count:0, preemption2-count:92  
iobound, pid:44, cpu-usage:164, user-cpu-usage:162, average-response-time:8, max-response-time:63, number-context-switches:191, preemption1-count:0, preemption2-count:86  
cpubound, pid:67, cpu-usage:111, user-cpu-usage:134, average-response-time:106, max-response-time:951, number-context-switches:98, preemption1-count:13, preemption2-count:84  
iobound, pid:6, cpu-usage:164, user-cpu-usage:165, average-response-time:8, max-response-time:20, number-context-switches:202, preemption1-count:0, preemption2-count:97  
iobound, pid:47, cpu-usage:165, user-cpu-usage:156, average-response-time:8, max-response-time:68, number-context-switches:202, preemption1-count:0, preemption2-count:97  
iobound, pid:54, cpu-usage:164, user-cpu-usage:159, average-response-time:8, max-response-time:79, number-context-switches:192, preemption1-count:0, preemption2-count:87  
iobound, pid:24, cpu-usage:166, user-cpu-usage:155, average-response-time:8, max-response-time:31, number-context-switches:192, preemption1-count:0, preemption2-count:86  
iobound, pid:11, cpu-usage:166, user-cpu-usage:161, average-response-time:7, max-response-time:19, number-context-switches:197, preemption1-count:0, preemption2-count:91  
iobound, pid:50, cpu-usage:163, user-cpu-usage:164, average-response-time:8, max-response-time:72, number-context-switches:207, preemption1-count:0, preemption2-count:103  
iobound, pid:39, cpu-usage:165, user-cpu-usage:166, average-response-time:8, max-response-time:55, number-context-switches:199, preemption1-count:0, preemption2-count:94  
iobound, pid:34, cpu-usage:163, user-cpu-usage:163, average-response-time:8, max-response-time:47, number-context-switches:203, preemption1-count:0, preemption2-count:99  
iobound, pid:53, cpu-usage:166, user-cpu-usage:159, average-response-time:8, max-response-time:77, number-context-switches:184, preemption1-count:0, preemption2-count:78  
iobound, pid:61, cpu-usage:133, user-cpu-usage:137, average-response-time:18, max-response-time:845, number-context-switches:176, preemption1-count:0, preemption2-count:91  
iobound, pid:33, cpu-usage:165, user-cpu-usage:155, average-response-time:8, max-response-time:46, number-context-switches:193, preemption1-count:0, preemption2-count:88

Figure 5: Benchmark D with starvation prevention enabled

```
#####
Greetings from Suraj Aralihalli!
Username: saraliha
#####

cpubound, pid:7, cpu-usage:1230, user-cpu-usage:1287, average-response-time:637, max-response-time:718, number-context-switches:14, preemption1-count:13, preemption2-count:0
cpubound, pid:8, cpu-usage:1230, user-cpu-usage:1200, average-response-time:644, max-response-time:720, number-context-switches:14, preemption1-count:13, preemption2-count:0
cpubound, pid:9, cpu-usage:1230, user-cpu-usage:1200, average-response-time:646, max-response-time:718, number-context-switches:14, preemption1-count:13, preemption2-count:0
cpubound, pid:10, cpu-usage:1230, user-cpu-usage:1200, average-response-time:647, max-response-time:718, number-context-switches:14, preemption1-count:13, preemption2-count:0
cpubound, pid:11, cpu-usage:1230, user-cpu-usage:1200, average-response-time:648, max-response-time:718, number-context-switches:14, preemption1-count:13, preemption2-count:0
cpubound, pid:2, cpu-usage:1332, user-cpu-usage:1300, average-response-time:599, max-response-time:718, number-context-switches:15, preemption1-count:14, preemption2-count:0
cpubound, pid:5, cpu-usage:1333, user-cpu-usage:1300, average-response-time:600, max-response-time:718, number-context-switches:15, preemption1-count:14, preemption2-count:0
cpubound, pid:6, cpu-usage:1333, user-cpu-usage:1300, average-response-time:601, max-response-time:718, number-context-switches:15, preemption1-count:14, preemption2-count:0
□
```

Figure 6: RT main(): Benchmark A without starvation prevention enabled

```
#####
Greetings from Suraj Aralihalli!
Username: saraliha
#####

cpubound, pid:8, cpu-usage:1230, user-cpu-usage:1207, average-response-time:450, max-response-time:646, number-context-switches:20, preemption1-count:19, preemption2-count:0
cpubound, pid:6, cpu-usage:1210, user-cpu-usage:1180, average-response-time:452, max-response-time:666, number-context-switches:20, preemption1-count:19, preemption2-count:0
cpubound, pid:5, cpu-usage:1271, user-cpu-usage:1240, average-response-time:375, max-response-time:1230, number-context-switches:24, preemption1-count:23, preemption2-count:0
cpubound, pid:7, cpu-usage:1323, user-cpu-usage:1290, average-response-time:427, max-response-time:687, number-context-switches:21, preemption1-count:20, preemption2-count:0
cpubound, pid:9, cpu-usage:1323, user-cpu-usage:1290, average-response-time:408, max-response-time:646, number-context-switches:22, preemption1-count:21, preemption2-count:0
cpubound, pid:10, cpu-usage:1230, user-cpu-usage:1200, average-response-time:413, max-response-time:646, number-context-switches:22, preemption1-count:21, preemption2-count:0
cpubound, pid:11, cpu-usage:1292, user-cpu-usage:1260, average-response-time:430, max-response-time:666, number-context-switches:21, preemption1-count:20, preemption2-count:0
cpubound, pid:2, cpu-usage:1353, user-cpu-usage:1320, average-response-time:409, max-response-time:697, number-context-switches:22, preemption1-count:21, preemption2-count:0
```

Figure 7: RT main(): Benchmark A with starvation prevention enabled

```
#####
Greetings from Suraj Aralihalli!
Username: saraliha
#####

cpubound, pid:8, cpu-usage:1230, user-cpu-usage:1207, average-response-time:450, max-response-time:646, number-context-switches:20, preemption1-count:19, preemption2-count:0
cpubound, pid:6, cpu-usage:1210, user-cpu-usage:1180, average-response-time:452, max-response-time:666, number-context-switches:20, preemption1-count:19, preemption2-count:0
cpubound, pid:5, cpu-usage:1271, user-cpu-usage:1240, average-response-time:375, max-response-time:1230, number-context-switches:24, preemption1-count:23, preemption2-count:0
cpubound, pid:7, cpu-usage:1323, user-cpu-usage:1290, average-response-time:427, max-response-time:687, number-context-switches:21, preemption1-count:20, preemption2-count:0
cpubound, pid:9, cpu-usage:1323, user-cpu-usage:1290, average-response-time:408, max-response-time:646, number-context-switches:22, preemption1-count:21, preemption2-count:0
cpubound, pid:10, cpu-usage:1230, user-cpu-usage:1200, average-response-time:413, max-response-time:646, number-context-switches:22, preemption1-count:21, preemption2-count:0
cpubound, pid:11, cpu-usage:1292, user-cpu-usage:1260, average-response-time:430, max-response-time:666, number-context-switches:21, preemption1-count:20, preemption2-count:0
cpubound, pid:2, cpu-usage:1353, user-cpu-usage:1320, average-response-time:409, max-response-time:697, number-context-switches:22, preemption1-count:21, preemption2-count:0
```

Figure 8: Benchmark A with starvation prevention enabled

```
#####
Greetings from Suraj Aralihalli!
Username: saraliha
#####

iobound, pid:10, cpu-usage:192, user-cpu-usage:185, average-response-time:0, max-response-time:9, number-context-switches:189, preemption1-count:0, preemption2-count:66
iobound, pid:11, cpu-usage:194, user-cpu-usage:149, average-response-time:0, max-response-time:11, number-context-switches:148, preemption1-count:0, preemption2-count:24
iobound, pid:6, cpu-usage:194, user-cpu-usage:159, average-response-time:0, max-response-time:3, number-context-switches:160, preemption1-count:0, preemption2-count:36
iobound, pid:5, cpu-usage:195, user-cpu-usage:124, average-response-time:0, max-response-time:1, number-context-switches:125, preemption1-count:0, preemption2-count:0
iobound, pid:7, cpu-usage:195, user-cpu-usage:124, average-response-time:0, max-response-time:4, number-context-switches:125, preemption1-count:0, preemption2-count:0
iobound, pid:8, cpu-usage:195, user-cpu-usage:127, average-response-time:0, max-response-time:6, number-context-switches:127, preemption1-count:0, preemption2-count:2
iobound, pid:2, cpu-usage:195, user-cpu-usage:128, average-response-time:0, max-response-time:3, number-context-switches:131, preemption1-count:0, preemption2-count:6
iobound, pid:9, cpu-usage:195, user-cpu-usage:127, average-response-time:0, max-response-time:7, number-context-switches:127, preemption1-count:0, preemption2-count:2
```

Figure 9: Benchmark B with starvation prevention enabled



```
#####
Greetings from Suraj Aralihalli!
Username: saraliha
#####

cpubound, pid:10, cpu-usage:2356, user-cpu-usage:2372, average-response-time:53, max-response-time:184, number-context-switches:147, preemption1-count:24, preemption2-count:122
cpubound, pid:9, cpu-usage:2293, user-cpu-usage:2305, average-response-time:51, max-response-time:85, number-context-switches:155, preemption1-count:24, preemption2-count:130
cpubound, pid:8, cpu-usage:2402, user-cpu-usage:2404, average-response-time:57, max-response-time:281, number-context-switches:138, preemption1-count:25, preemption2-count:112
cpubound, pid:11, cpu-usage:2393, user-cpu-usage:2401, average-response-time:53, max-response-time:151, number-context-switches:148, preemption1-count:25, preemption2-count:122
iobound, pid:5, cpu-usage:195, user-cpu-usage:124, average-response-time:0, max-response-time:1, number-context-switches:125, preemption1-count:0, preemption2-count:0
iobound, pid:2, cpu-usage:195, user-cpu-usage:126, average-response-time:0, max-response-time:1, number-context-switches:128, preemption1-count:0, preemption2-count:3
iobound, pid:6, cpu-usage:195, user-cpu-usage:127, average-response-time:0, max-response-time:3, number-context-switches:126, preemption1-count:0, preemption2-count:1
iobound, pid:7, cpu-usage:195, user-cpu-usage:125, average-response-time:0, max-response-time:4, number-context-switches:126, preemption1-count:0, preemption2-count:1
```

Figure 10: Benchmark C with starvation prevention enabled