Online on xv6 - Scheduler

Section: B1 Time: 30 minutes

Your task is to implement the FCFS scheduling algorithm in xv6. The default RR scheduling will work only for the processes with pid 1 (init) and 2(sh). For all other processes, whichever comes first, you will need to run it till completion.

You are provided a user program `testloop.c`. It simulates a long running task by running a loop. The iteration count is provided as an argument. Examine its contents to understand how it works.

You will need to modify the `scheduler()` function in `proc.c`. Feel free to add any helper fields in the proc structure.

Input:

```
testloop 150 &;
testloop 100 &;
testloop 50 &;
ls
```

Output:

```
Process 5: Starting 150 iterations at time 53
Process 5: Finished at time 156
Process 8: Starting 100 iterations at time 157
Process 8: Finished at time 224
Process 11: Starting 50 iterations at time 224
Process 11: Finished at time 258
<output of ls, omitted for brevity>
```

Note:

Set `CPUS := 1` in the Makefile. You must provide the input in the shell one by one (not all at once). The order of your output must match with the given order.

You will also need to ensure that the first two processes (pid 1 and 2) are not affected by the FCFS algorithm. These should run in the default RR algorithm.

Submission:

```
git add --all
git diff HEAD > ../2005010.patch
```