Lab 11

Github: steinhs

Source Code

FCFS

```
void first_come_first_served(){
    /* Your code for FCFS algorithm here */
    int x = 1, y = 0, previousEndtime = 0;
    while (x!=0){

        if (previousEndtime > proc[y].arrivaltime){
            proc[y].starttime = previousEndtime;
            proc[y].endtime = previousEndtime + proc[y].runtime;
        } else {
            proc[y].starttime = proc[y].arrivaltime;
            proc[y].endtime = proc[y].arrivaltime + proc[y].runtime;
        }

        printf("Process %d started at time %d and finished at time
%d\n", proc[y].id, proc[y].starttime, proc[y].endtime);

        previousEndtime = proc[y].endtime;

        y++;
        x = proc[y].id;
    }
}
```

SJF

Also implemented function cmp() for qsort which is included below shortest_job_first().

```
void shortest_job_first(){
    /* Your code for SJF algorithm here */
    qsort (proc, 100, sizeof(process), cmp);
    int x = 1, y = 0, previousEndtime = 0;
    for (y = 0; y < 100; ++y) {

        if (proc[y].id != 0) {
            if (previousEndtime > proc[y].arrivaltime){
                proc[y].starttime = previousEndtime;
                proc[y].endtime = previousEndtime + proc[y].runtime;
            } else {
                proc[y].starttime = proc[y].arrivaltime;
                proc[y].endtime = proc[y].arrivaltime +
proc[y].runtime;
            }
            printf("Process %d started at time %d and finished at
time %d\n", proc[y].id, proc[y].starttime, proc[y].endtime);

            previousEndtime = proc[y].endtime;
        }
    }
}


int cmp(const void *p1, const void *p2)
{
    const struct process *el1 = p1;
    const struct process *el2 = p2;

    if (el1->runtime < el2->runtime)
        return -1;
    else if (el1->runtime > el2->runtime)
        return 1;
    else
        return 0;
}
```

Output

```
$ ./scheduling
Process  arrival  runtime
1   0   4
2   7   10
3   9   2
4   13  8
5   25  1
6   35  7
7   36  2
First come first served...
Process 1 started at time 0 and finished at time 4
Process 2 started at time 7 and finished at time 17
Process 3 started at time 17 and finished at time 19
Process 4 started at time 19 and finished at time 27
Process 5 started at time 27 and finished at time 28
Process 6 started at time 35 and finished at time 42
Process 7 started at time 42 and finished at time 44
Shortest job first...
Process 5 started at time 25 and finished at time 26
Process 3 started at time 26 and finished at time 28
Process 7 started at time 36 and finished at time 38
Process 1 started at time 38 and finished at time 42
Process 6 started at time 42 and finished at time 49
Process 4 started at time 49 and finished at time 57
Process 2 started at time 57 and finished at time 67
```