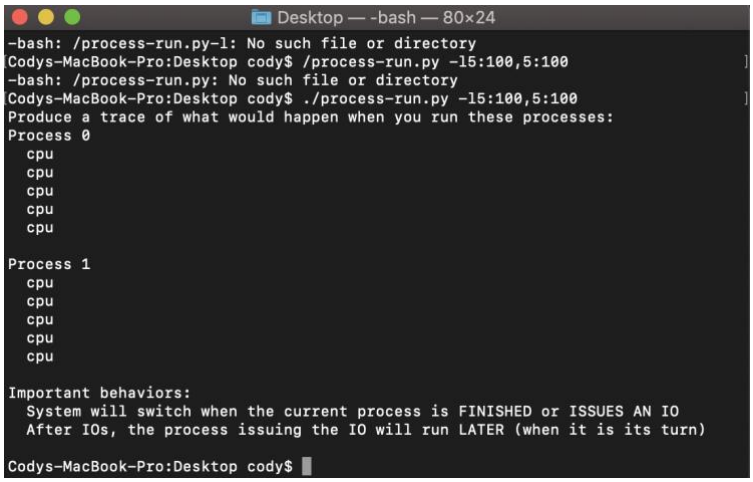
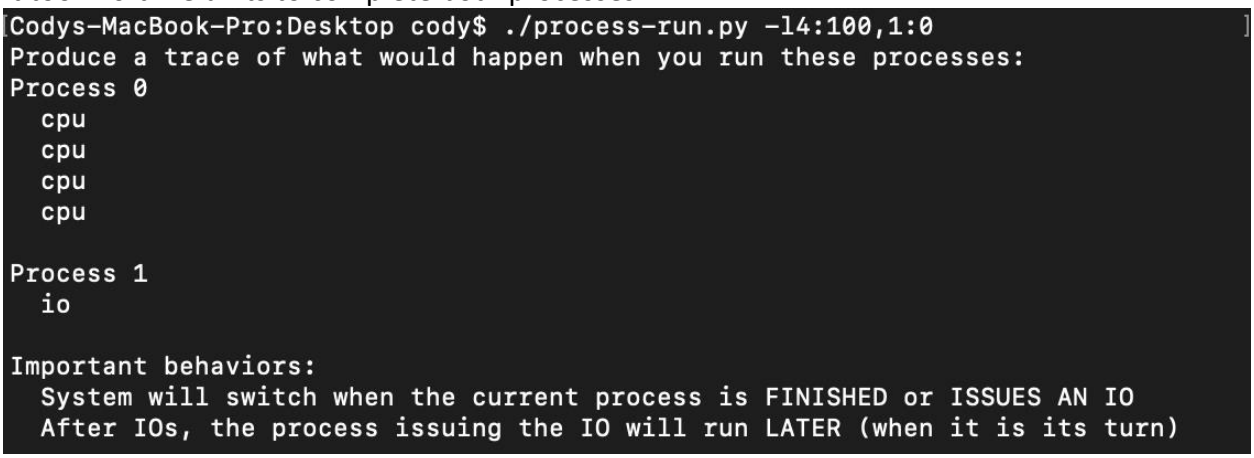
Cody Carroll
Simulation Assignment 1
Fall 2018

1) The CPU utilization should be at 100% because the process we specified is "5:100,5:100" which means it should consist of 5 instructions per process, and the chances that each instruction is a CPU instruction are 100%; which means they are going to all be CPU instructions putting it at 100% utilization.



2) It took 10-time units to complete both processes.



3) When you switch the order of the processes they both fun at the same time. Yes, switching the order matters because since the processes run at the same time the total time is only 6-time units instead of 10 times units before they were switched.

```
Codys-MacBook-Pro:Desktop cody$ ./process-run.py -l1:0,4:100
Produce a trace of what would happen when you run these processes:
Process 0
  io

Process 1
  cpu
  cpu
  cpu
  cpu

Important behaviors:
  System will switch when the current process is FINISHED or ISSUES AN IO
  After IOs, the process issuing the IO will run LATER (when it is its turn)
```

4) In the i/o process the time spent was 9 and, in the CPU process the time spent was 10. Because there is an i/o initialization step, the time is reduced to 9-time units. If the process finish the CPU tasks first then move to the IO tasks, the time is 10-time units.

```
Codys-MacBook-Pro:Desktop cody$ ./process-run.py -l1:0,4:100 -c -S SWITCH_ON_END
Time     PID: 0     PID: 1        CPU        IOs
  1      RUN:io      READY          1
  2     WAITING      READY                    1
  3     WAITING      READY                    1
  4     WAITING      READY                    1
  5     WAITING      READY                    1
  6*       DONE    RUN:cpu          1
  7        DONE    RUN:cpu          1
  8        DONE    RUN:cpu          1
  9        DONE    RUN:cpu          1
Codys-MacBook-Pro:Desktop cody$ ./process-run.py -l4:100,1:0 -c -S SWITCH_ON_END
Time     PID: 0     PID: 1        CPU        IOs
  1     RUN:cpu      READY          1
  2     RUN:cpu      READY          1
  3     RUN:cpu      READY          1
  4     RUN:cpu      READY          1
  5        DONE     RUN:io          1
  6        DONE    WAITING                  1
  7        DONE    WAITING                  1
  8        DONE    WAITING                  1
  9        DONE    WAITING                  1
 10*       DONE       DONE
```

5) When you use SWITCH_ON_IO, the time is different because the two processes run at the same time instead of separately. The IO doesn't have to be complete before the CPU processes are ran.

```
Codys-MacBook-Pro:Desktop cody$ ./process-run.py -l1:0,4:100 -c -S SWITCH_ON_IO
Time     PID: 0     PID: 1        CPU        IOs
  1      RUN:io      READY          1
  2     WAITING    RUN:cpu          1          1
  3     WAITING    RUN:cpu          1          1
  4     WAITING    RUN:cpu          1          1
  5     WAITING    RUN:cpu          1          1
  6*       DONE       DONE
Codys-MacBook-Pro:Desktop cody$ ./process-run.py -l4:100,1:0 -c -S SWITCH_ON_IO
Time     PID: 0     PID: 1        CPU        IOs
  1     RUN:cpu      READY          1
  2     RUN:cpu      READY          1
  3     RUN:cpu      READY          1
  4     RUN:cpu      READY          1
  5        DONE     RUN:io          1
  6        DONE    WAITING                    1
  7        DONE    WAITING                    1
  8        DONE    WAITING                    1
  9        DONE    WAITING                    1
 10*       DONE       DONE
```