Testing

Hypothesis:

I can measure the yield time for every single process. lp can be run as command in xv6 to see running processes. I believe that I can measure the time the process takes before it yields by saving the information to the proc structure.

Changes I made:

I added different system calls to measure the statistics of the process which measures the exit time of a process along with yield time. Several additions have been made to proc.h to keep track of additional clock ticks, creation time, end time and yield time. When a process is exiting i.e. wait() or kill() is called, I made it display the name, pid, creation and yield time so I can keep track of how long the process is taking.

Example:

We can see that we passed 20 as clock tick value to the xv6 and we notice the ytime for pname: lp to be 20 more the ctime. This leads me to believe that my scheduler is working as expected and rescheduling and same job again.

```
$S20 alsoNice 20 lp
Name     pid     state   creationtime    yieldtime
 init    1       SLEEPING        0       0
sh       2       SLEEPING        5       0
sh       11      SLEEPING        30452   0
lp       12      RUNNING         30453   0

 -------------------------------------------------

 pname: lp       pid: 12         ctime: 30453    ytime: 30473

 -------------------------------------------------

 -------------------------------------------------

 pname: sh       pid: 11         ctime: 30452    ytime: 0

 -------------------------------------------------
```