1 Project Report [25 pts.]

1.1 Assignment Description [1 pt.]

1.2 Assignment Deliverables [5 pts.]

See example report.

1.3 Implementation [5 pts.]

- File and line numbers need to be correct. Each system call implementation must, at a minimum, *list all the files, together with line numbers*, where modifications were made to insert the new system call.
- Writeup will likely closely follow example report.

1.4 Required Tests [13 pts.]

Rewrite for test-expected output-results-discussion format.

- System call tracing correctness [3 pt.]
- Date system call and command [5 pt.]
- Elapsed time calculation correctness [3 pt.]
- ctrl – p modifications [2 pt.]

2 Project Code [25 pts.]

Note: Student code not submitted cannot be graded – no late work. An improperly submitted project is considered to be not submitted. Students should follow the submission guidelines from the Survival Guide. **Students are responsible for correctly submitting their projects.**

2.1 Conditional Compilation [4 pts.]

- Must include #ifdef statements for each piece of new code except for files `usys.S`, `user.h`, and `syscall.h` which do not use conditional compilation.
- Project must compile correctly *with the conditional compilation flag both turned on and turned off.*

2.2 System Call Tracing [5 pts.]

- Must print system call name, not number.
- Must make a minimal change to the existing `syscall()` routine.
- Must use an array similar to the existing function dispatch table.
- Other errors [-1 pt. for each]

2.3 Date System Call [5 pts.]

- Must use use `argptr()` to extract the `struct rtcdate` pointer.
- Calls `cmostime()` correctly.
- Must implement system call correctly per project description.

Date Command [5 pt.]

- Must print date information per the assignment description.

2.4 ctrl – p Modifications [6 pts.]

- Initializes `start_ticks` correctly.
- Must print an appropriate header in `procdump()` using `cprintf()`
- Time output must be to the nearest hundreth of a second.