CODE MODIFICATION REPORT

Makefile

- For compilation test :

```
CS333_PROJECT ?= 1
PRINT_SYSCALLS ?= 0
```

- For system call tracing :

```
CS333_PROJECT ?= 1
PRINT_SYSCALLS ?= 1
```

- For date system call:

```
CS333_UPROGS += _date
```

Task 1

- Syscall.c

```
#ifdef PRINT_SYSCALLS
cprintf ("%s -> %d \n", syscallnames[num], syscalls[num]());
#endif
```

Task 2

- User.h

```
#ifdef CS333_P1
int date(struct rtcdate*);
#endif // CS333_P1
```

- Usys.S

```
SYSCALL(date)
```

- Syscall.h

```
#define SYS_date    SYS_halt+1
```

- Syscall.c

```c
#ifdef CS333_P1
extern int sys_date(void);
#endif // CS333_P1
#ifdef CS333_P1
[SYS_date]   sys_date,
#endif // CS333_P1
```

```c
#ifdef CS333_P1
[SYS_date]   "date",
#endif // CS333_P1
```

- Sysproc.c

```c
int
sys_date(void)
{
  struct rtcdate *d;
  if(argptr(0, (void*)&d, sizeof(struct rtcdate))<0)
          return -1;
cmostime(d);
return 0;
}
```

- Date.c

```c
// r.hour %= 12;
// if (r.hour == 0) r.hour = 12;

printf(1, "%s %s  %d %s%d:%s%d:%s%d UTC %d\n", days[day], months[r.month], r.day,
PAD(r.hour), r.hour, PAD(r.minute), r.minute, PAD(r.second), r.second, r.year, s)
;
```

Task 3

- Proc.h

```c
uint start_ticks;
```

- Proc.c

```c
p->start_ticks = ticks;
```

```c
int time_sec;
  int time_msec;

  time_msec = ticks - p->start_ticks;
  time_sec = time_msec / 1000;
  time_msec = time_msec % 1000;

  char* zero = "";
  if(time_msec < 100 && time_msec >= 10)
    zero = "0";
  if(time_msec < 10)
    zero = "00";

  cprintf("%d\t%s\t\t%s%d.%s%d\t%s\t%d\t",
    p->pid, p->name, "      ", time_sec, zero, time_msec, states[p->state], p-
>sz);
```