Code Modification Assignment 1

Adi Rahman (1313618037)

Makefile

Line 3-4

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

syscall.c

Line 111-113

```
#ifdef CS333_P1
  extern int sys_date(void);
#endif //CS333_P1
```

Line 142-145

```
#ifdef CS333_P1
  [SYS_date]  sys_date,
#endif //CS333_P1
};
```

Line 184-188

```
#ifdef CS333_P1
    #ifdef PRINT_SYSCALLS
      cprintf("%s -> %d \n",syscallnames[num],num);
    #endif
  #endif //CS333_P1
```

syscall.h

Line 27

```
#define SYS_date  SYS_halt+1 // CS333_P1
```

user.h

Line 48-50

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

usys.S

Line 33

```
SYSCALL(date)
```

sysproc.c

Line 102-113

```
#ifdef CS333_P1

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

#endif //CS333_P1
```

proc.h

Line 54-56

```
#ifdef CS333_P1
  uint time_elapse;
  #endif // CS333_P1
```

proc.c

Line 153-155

```
#ifdef CS333_P1
    p -> time_elapse = ticks;
```

```
  #endif // CS333_P1
```

Line 569-607

```
procdumpP1(struct proc *p, char *state_string)
{
  char *state;
  uint numberState = p -> state;

  switch (numberState)
  {
  case 0:
    state="unused";
    break;
  case 1:
    state="embryo";
    break;
  case 2:
    state="sleep";
    break;
  case 3:
    state="runnable";
    break;
  case 4:
    state="run";
    break;
  default:
    state="zombie";
    break;
  }

  uint getElapsed = ticks-p->time_elapse;
  //Convert to second
  uint left = (getElapsed) / 1000;
  uint right = getElapsed % (left*1000);



  //cprintf("%d", secElapsed);

  cprintf("\n%d\t%s\t%d.%d\t%s\t%d\t", p->pid, p->name, left, right, state, p-
>sz);

}
```