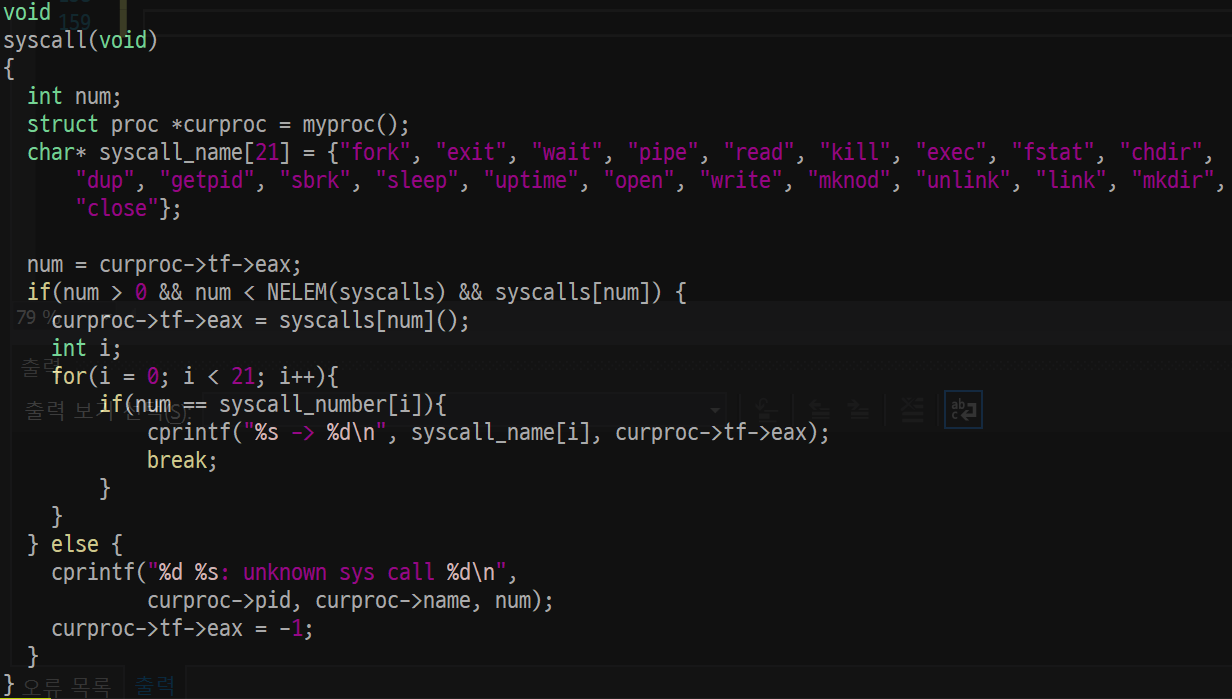
**Homework 3**

System Programming (12506)

2017029807 성창호

**Part 1.**

캡처 사진



**Text**

void

syscall(void)

{

int num;

struct proc \*curproc = myproc();

char\* syscall\_name[21] = {"fork", "exit", "wait", "pipe", "read", "kill", "exec", "fstat", "chdir",

"dup", "getpid", "sbrk", "sleep", "uptime", "open", "write", "mknod", "unlink", "link", "mkdir",

"close"};

num = curproc->tf->eax;

if(num > 0 && num < NELEM(syscalls) && syscalls[num]) {

curproc->tf->eax = syscalls[num]();

int i;

for(i = 0; i < 21; i++){

if(num == syscall\_number[i]){

cprintf("%s -> %d\n", syscall\_name[i], curproc->tf->eax);

break;

}

}

} else {

cprintf("%d %s: unknown sys call %d\n",

curproc->pid, curproc->name, num);

curproc->tf->eax = -1;

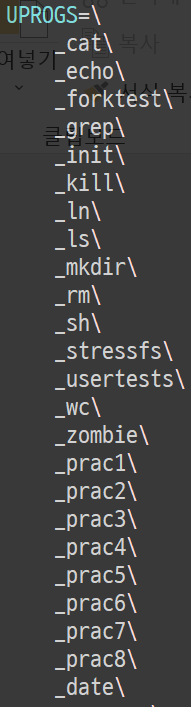
}

}

**Part 2.**

**Makefile**

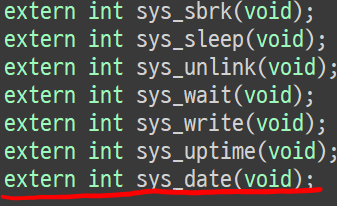
UPROGS에 \_date\를 다음과 같이 추가한다.



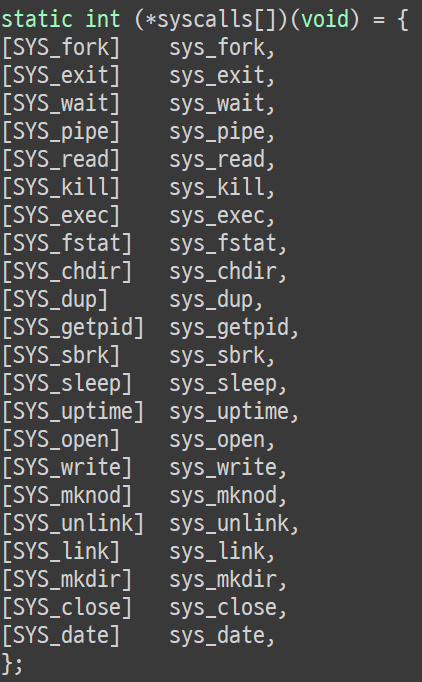


**syscall.c**

sys\_\*(void) 형식으로 시스템콜 함수를 선언하는 부분에 sys\_date함수를 다음과 같이 선언한다.



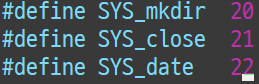
그리고 static int (\*syscall[])(void)에다가 [SYS\_date] sys\_date를 추가한다.





**syscall.h**

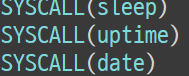
#define SYS\_date 22를 추가해준다.





**usys.S**

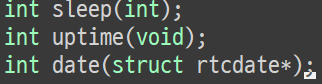
SYSCALL(date)를 추가해준다.





**user.h**

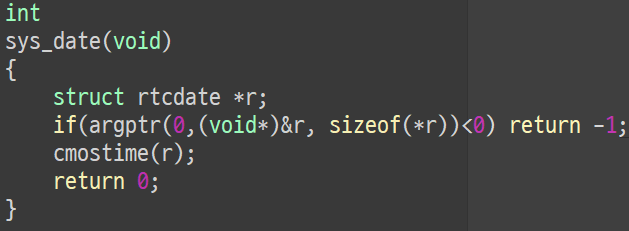
date 함수의 선언부를 작성한다. 인자는 struct rtcdate의 포인터이다.





**sysprog.c**

argptr 함수를 이용해서 struct rtcdate 포인터인 r이 유효한지 확인하고 유효할 경우 cmostime함수를 호출시켜 r에 시간 정보를 담는다.



**date.c**

