HA1_12340220_part-2

Step 1: Update the process structure

File: `proc.h`
Code Added:

```
// Add this inside struct proc
int userflag; // user-defined integer flag for each process
```

Step 2: Assign syscall numbers

File: `syscall.h`
Code Added:

```
#define SYS_setflag  26
#define SYS_getflag  27
```

Step 3: Declare handlers and update syscall table

File: `syscall.c`
Code Added:

```
// Added extern declarations at top
extern int sys_setflag(void);
extern int sys_getflag(void);

// Added into syscall table
[SYS_setflag] sys_setflag,
[SYS_getflag] sys_getflag,
```

Step 4: Implement the kernel handlers

File: `sysproc.c`
Code Added:

```
int sys_setflag(void) {
    int val;
    if (argint(0, &val) < 0)
        return -1;
    myproc()->userflag = val;
    return 0;
}

int sys_getflag(void) {
    return myproc()->userflag;
}
```

Step 5: Update user prototypes

File: `user.h`
Code Added:

```
int setflag(int value);
int getflag(void);
```

Step 6: Add syscall assembly stubs

File: `usys.S`
Code Added:

```
SYSCALL(setflag)
SYSCALL(getflag)
```

Step 7: User program example

File: `testflags.c`
Code Added:

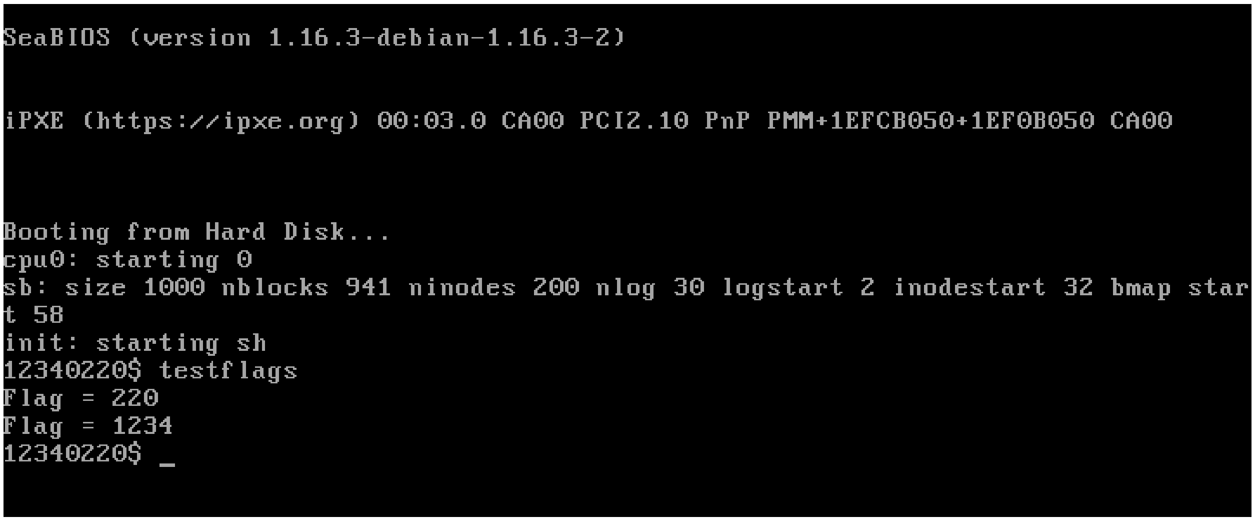
```
#include "types.h"
#include "stat.h"
#include "user.h"

int main() {
    setflag(220); // last 4 digits of roll number, dropped the initial 0
    printf(1, "Flag = %d\n", getflag());

    setflag(9999);
    printf(1, "Flag = %d\n", getflag());

    exit();
}
```

Step 8: Add to Makefile

File: `Makefile`
Code Added in UPROGS line:

_testflags\

Step 9: Output Screenshot



```
SeaBIOS (version 1.16.3-debian-1.16.3-2)


iPXE (https://ipxe.org) 00:03.0 CA00 PCI2.10 PnP PMM+1EFCB050+1EF0B050 CA00



Booting from Hard Disk...
cpu0: starting 0
sb: size 1000 nblocks 941 ninodes 200 nlog 30 logstart 2 inodestart 32 bmap star
t 58
init: starting sh
12340220$ testflags
Flag = 220
Flag = 1234
12340220$ _
```

Example:

$ testflags
Flag = 220
Flag = 9999
$