Step 1:

Code added in proc.h:

```
int userflag;
```

Step 2:

Code added in syscall.h:

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

Step 3:

Code added in syscall.c, added extern declarations:

```
extern int sys_setflag(void);
extern int sys_getflag(void);
```

Code added in syscall.c, added to syscall table:

```
[SYS_setflag] sys_setflag,
[SYS_getflag] sys_getflag,
```

Step 4:

Code added in sysproc.c:

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

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

Step 5:

Code added in user.h:

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

Step 6:

Code added in usys.S:

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

Step 7:

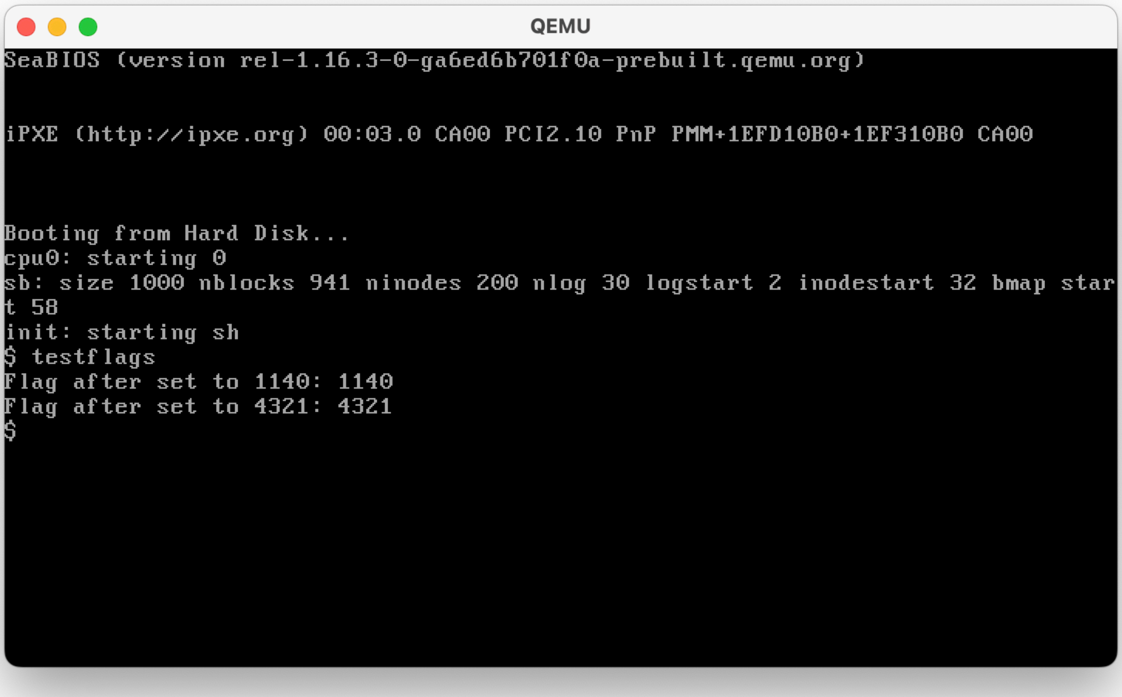
Created a new testflags.c file with the following code:

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

int main(void) {
    setflag(1140);
    printf(1, "Flag after set to 1140: %d\n", getflag());

    setflag(4321);
    printf(1, "Flag after set to 4321: %d\n", getflag());

    exit();
}
```

The Final step:

In Makefile, added the following line in UPGROS:

```
    _testflags\
```



```
SeaBIOS (version rel-1.16.3-0-ga6ed6b701f0a-prebuilt.qemu.org)


iPXE (http://ipxe.org) 00:03.0 CA00 PCI2.10 PnP PMM+1EFD10B0+1EF310B0 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
$ testflags
Flag after set to 1140: 1140
Flag after set to 4321: 4321
$
```

Screenshot of the final output in QEMU terminal
1140 are the last four digits of my ID