OS Project #1

B03901116 王彥稀, B03901018 楊程皓

April 7, 2018

Screenshots from our two versions of kernel:

```c
#include<sys/syscall.h>
#include<unistd.h>
#include<stdio.h>
int main() {
        long ret1, ret2;
        syscall(337);
        ret1 = syscall(338, (long)13, (long)25);
        ret2 = syscall(339, (long)18, (long)116);
        printf("%ld %ld\n", ret1, ret2);
        return 0;
}
```

Figure 1: Testing program for kernel version 1.

```
danny@danny-VirtualBox:~$ gcc test.c -o test
danny@danny-VirtualBox:~$ ./test
325 18
danny@danny-VirtualBox:~$ dmesg | tail -n 1
[  328.126414] HELLO SYSTEM CALL B03901018 B03901116
```

Figure 2: Multiply 13 and 25, take min of 18 and 116 and hello system call to verify correctness.

```
harvey@harvey-VirtualBox:~/Desktop/OS_2018/Project1$ g++ test.cpp -o test
harvey@harvey-VirtualBox:~/Desktop/OS_2018/Project1$ ./test
10000 tests of mul, min correct.
harvey@harvey-VirtualBox:~/Desktop/OS_2018/Project1$ dmesg | tail -n 1
[ 2628.983994] HELLO SYSTEM CALL Student-b03901018 Student-b03901116
```

Figure 3: Testing program for kernel version 2. Run 10000 testcases of multiplication and min operations.

Add sys_hello, sys_multiply, sys_min as instructed in slides (with different input variables). In test file (test.cpp), call hello, multiply (0-99, 0-99), and find the minimum of (0-99, 0-99), total 10000 tests. All of the system calls get the correct returns and outputs.

The modified files (Makefile, hello.c, syscall_table_32.S, syscalls.h unistd_32.h) are included. Our test.cpp (10000 trials) for testing is also included.

Faces difficulties: Use type ”long long” in the testing program at the beginning, and found the answer is wrong.