**getCallCount() MANUAL PAGE**

**NAME**

**getCallCount –** getting the number of times a specific system call has been invoked

**SYNOPSIS**

#include <user.h>

int **getCallCount**(int);

**DESCRIPTION**

**getCallCount** is the system call that will shows the number of time a system call has been executed. User passes to **getCallCount()** an integer indicates the system call he/she wants to check. However, the user cannot pass values that are:

* Non-integer
* Different from numbers that has been assign to the list of system calls

( Only from 1-> 22 are allowed)

**RETURN VALUES**

Upon successful complement, **getCallCount(int)** will return the number of times a system call has been executed. Otherwise, a value of -1 will be return if the value of the system call passed to the function is invalid.

**ERRORS**

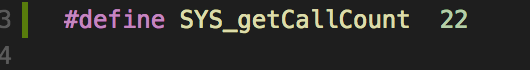
If the argument pass to the call is an integer that is not in the range 1->22, the function would return -1 to indicate an invalid argument

**Implementing getCallCount()**

To get getCallCount() running, I change a total of 7 files in xv6.

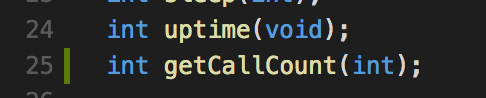
* **syscall.h:**

Set a specific number associated with the new system call (22)



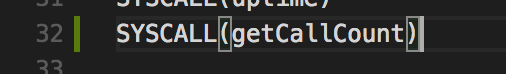
* **user.h:**

Specify a new function signature so that the program would know **getCallCount()** is a system call



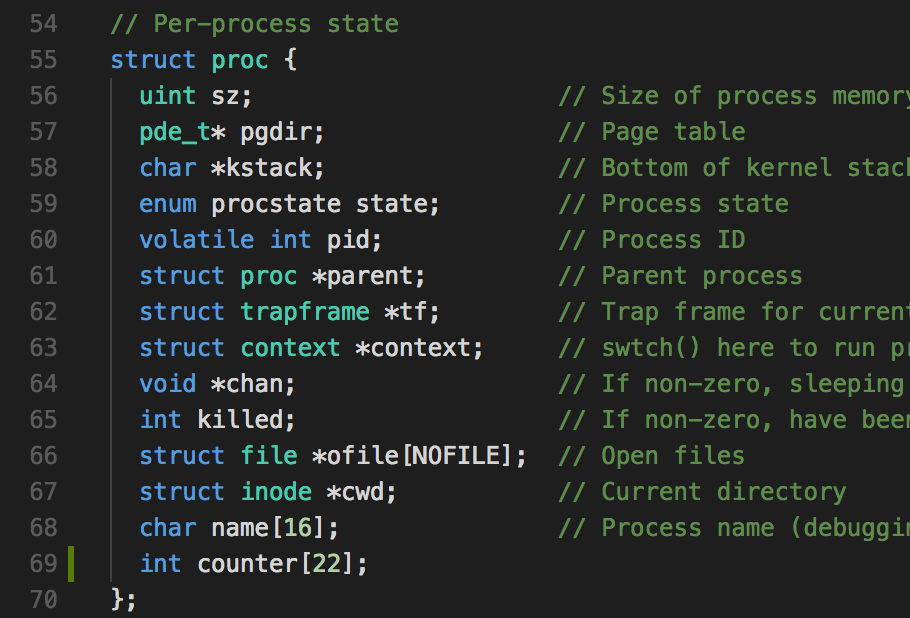
* **usys.S:**

Add a new system call so that the program would know how to get the number associated with the getCallCount system call to procede with the program.



* **proc.h:**

Add a counter array for the proc struct so that every time a process run, the counter array will keep track of how many time a system call has been executed.



* **proc.c:**

Reset the counter array to 0 when creating a new process. This is in the function **allocproc().**



* **syscall.c:**

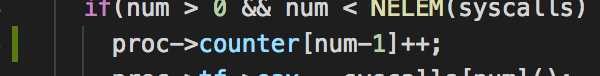
In **syscall.c,** an extern int sys\_getCallcount(void) would be added to the list of system calls. It would help the program to able to get the **sys\_getCallCount()** function.

../../Desktop/Screen%20Shot%202018-02-28%20at%205.39.21%20PM.png

Then, the array **syscalls** will add a map from SYS\_getCallCount number to the function sys\_getCallCount so that the system call can be executed in the **syscall()** function.

../../Desktop/Screen%20Shot%202018-02-28%20at%205.39.27%20PM.png

Every time a system call got executed, the **counter** item at index **num** (which is the number associated with the executed system call) will be incremented by 1.



* **sysproc.c:**

Implement the getCallCount() system call in the function **sys\_getCallCount().** This function will return the result of the system call.

