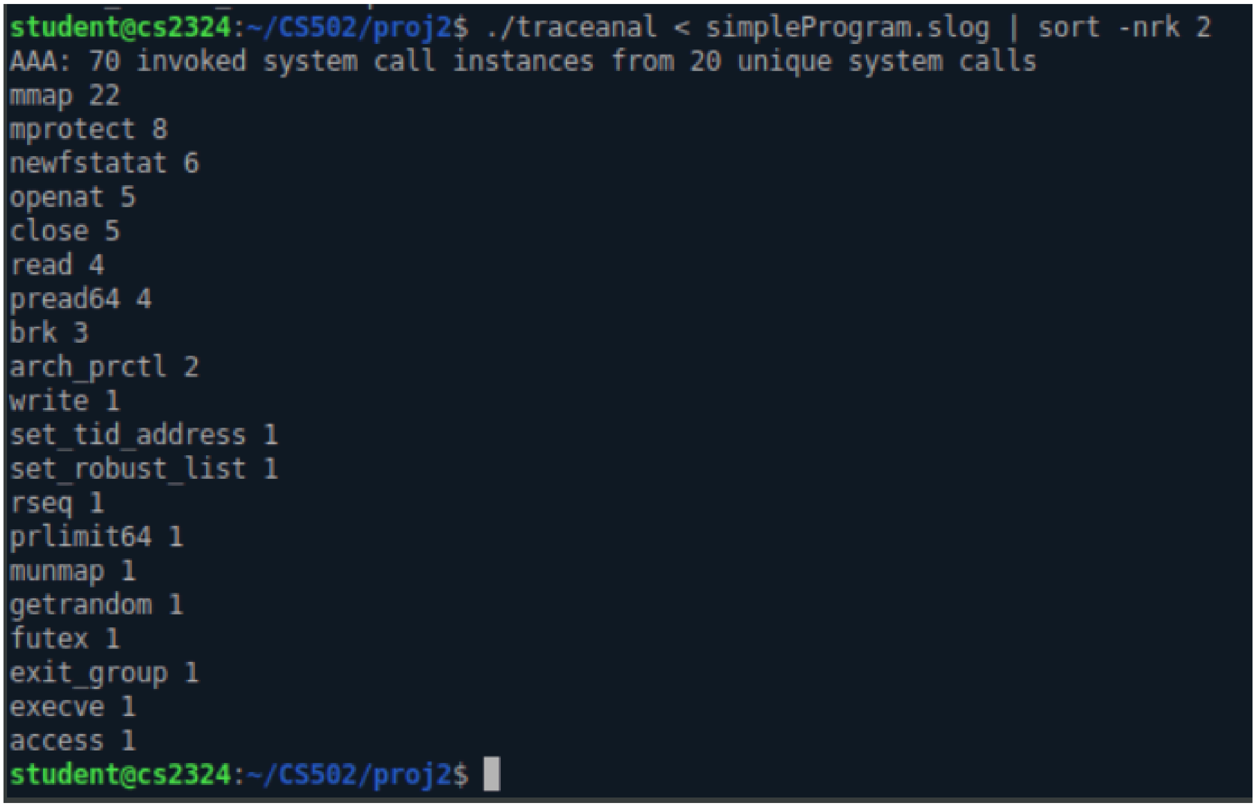
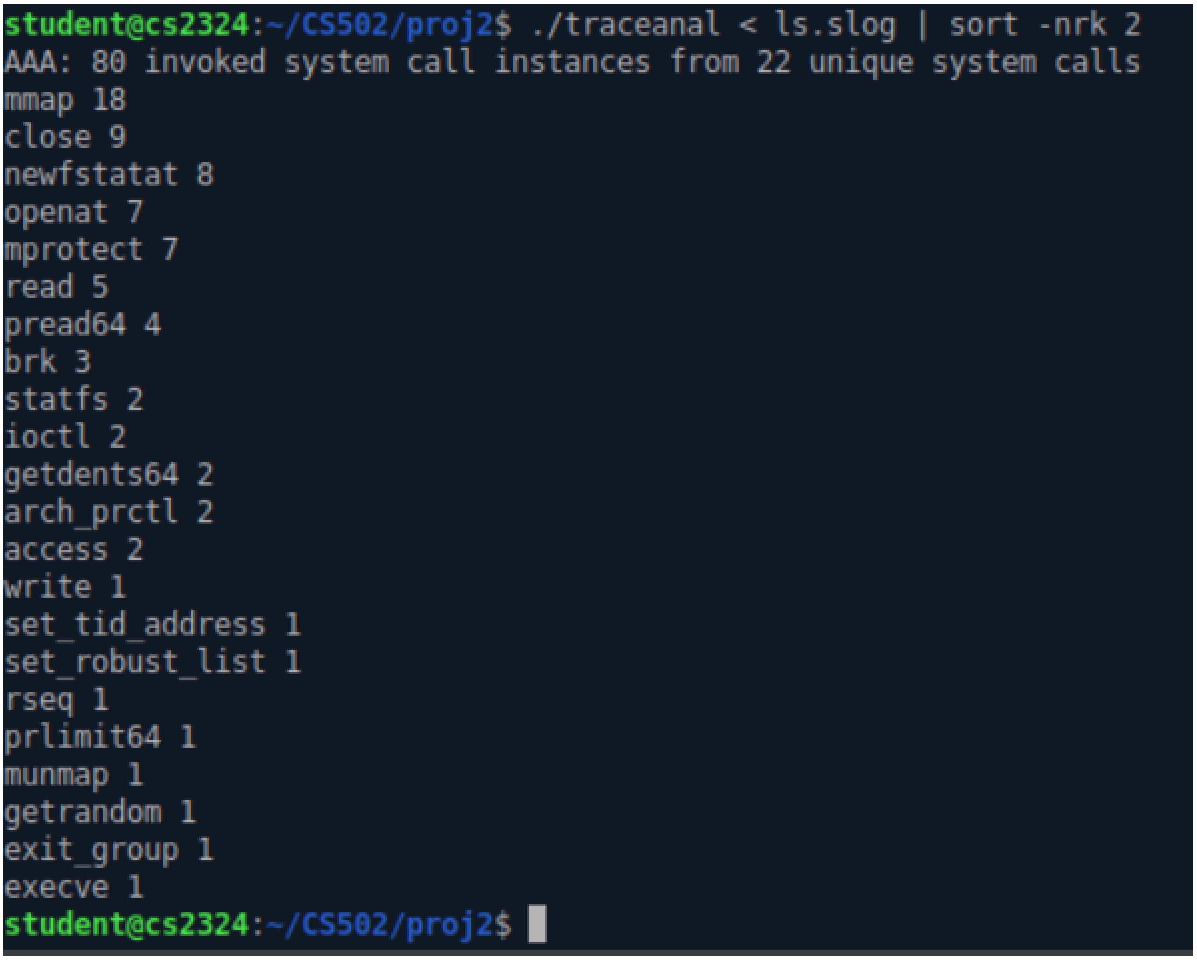
1. Observation of program execution behaviors shows that many system calls are invoked as part of starting up a program. To examine this start-up behavior, construct a simple program that makes no system calls and analyze it using traceanal. Do all programs exhibit a similar start-up behavior in terms of which system calls are used and their relative sequence?

To examine this behavior, I've created a program called 'simpleProgram.cpp' and I utilized the 'dostrace' executable to generate a strace log file from the program. The result of running 'traceanal' on 'simpleProgram' is the following:



```
student@cs2324:~/CS502/proj2$ ./traceanal < simpleProgram.slog | sort -nrk 2
AAA: 70 invoked system call instances from 20 unique system calls
mmap 22
mprotect 8
newfstatat 6
openat 5
close 5
read 4
pread64 4
brk 3
arch_prctl 2
write 1
set_tid_address 1
set_robust_list 1
rseq 1
prlimit64 1
munmap 1
getrandom 1
futex 1
exit_group 1
execve 1
access 1
student@cs2324:~/CS502/proj2$
```

It appears that the start-up behavior of this program, which only prints out the message "Hello", is very similar to the behavior seen in the 'ls.slog' file and it's easier to see the similarity by displaying the output of the same command, except that the log file has been changed to 'ls.slog':



```
student@cs2324:~/CS502/proj2$ ./traceanal < ls.slog | sort -nrk 2
AAA: 80 invoked system call instances from 22 unique system calls
mmap 18
close 9
newfstatat 8
openat 7
mprotect 7
read 5
pread64 4
brk 3
statfs 2
ioctl 2
getdents64 2
arch_prctl 2
access 2
write 1
set_tid_address 1
set_robust_list 1
rseq 1
prlimit64 1
munmap 1
getrandom 1
exit_group 1
execve 1
student@cs2324:~/CS502/proj2$ 
```

The behavior of both programs is very similar with nearly identical numbers of calls to many of the same system calls.

2. Researchers have proposed using the system call sequence of a program as a “signature” for that program as a means to detect if a copy of a program is substituted by an intruder. Investigate the validity of this idea by checking if the signature of different executions of the same program are the same. The particular counts of system calls may vary, but are the sequences similar? What if different command line arguments are used for a command? Is there variation in the sequence? Does the sequence change if the amount of data or duration of execution varies for a program?

To verify this proposal, I ran the ‘simpleProgram’ executable from the last question three times and recorded the strace log file for each run in a folder called ‘Question2Runs’. From what I could see in the log file for each run, it appears that the system calls used by the program in each run are exactly the same and only the addresses that are used differ between each run.

3. How much variation and commonality do you observe from invocations of different commands? You should try to separate out the start-up behavior common to all commands and the command-specific portion.

There does not appear to be much variation between the invocations of different commands since the start-up behavior appears to stay similar across many of the commands. I’ve created log files using strace for ‘ls’, ‘pwd’, and the simple program I created for Question 1. In each of these log files, the layout of many of the system calls is quite similar in the beginning, although there could be slight differences with regards to the total number of each system call used. While the start-up behavior shares similarities between the different commands, the command-specific portion does change, which I could see by comparing the simpleProgram.slog, pwd.slog, and ls.slog files that I’ve created with strace. Once we get to the command-specific portion of the log files, the types of system calls and number of those calls used vary significantly between the different commands.