***Operating Systems***

***Homework 1***

Antonia Savu

Problem 1.1: library and system calls

1. A picture containing text

   Description automatically generatedText

   Description automatically generated with medium confidenceThe *date* function uses **43 library calls** and **40 system calls**. If we call strace/ltrace with the option -c, they get displayed in a more compact manner. In order to be able to do this, I used a Debian WSL, as other Unix distributions do not display the result of ltrace properly.
2. The most frequent system calls are:

-*mmap:* it is used to map files into memory; more specifically, it creates a new mapping in the virtual address space of the process

-*close:* simply closes a file descriptor once the process terminates

-*fstat:* obtains the required information about a file

The most frequent library calls are:

-*fwrite:* writes data from an array to a stream using buffering, and then calls write().

-*fputc:* writes a single character to the stream

-*\_\_freading:*  basic function that indicates if the file is in read-only, or if the last function on the stream was a read operation (such as *fread)*. If that is the case, it returns non-zero; otherwise, zero.

Problem 1.2: system call errors

1. One of the most common errors associated with the *open()* function is when the file or path inside of the function doesn’t exist; also known as *FileNotFoundError: [Errno 2] No such file or directory*

As for the *close()* function, an error will occur when the file is not currently(or it never was) open.

1. When there are no errors, the value of *errno* is 0.