

# Lab 1

Daniel O'Brien

September 0th, 2019

## **1 What are the advantages and disadvantages of using the same system call interface for manipulating both files and devices?**

The advantages of creating a system-call interface for manipulating both files and devices come as a result of the abstraction layer it provides for developers. It treats devices similar to files, and lets you read/write to them. As long as the developer knows how to utilize the system-call interface, they can create a driver program for their device that will allow the program to run on any system using that same interface. This saves time because a developer is able to support their programs on multiple machines without having to worry about compatibility issues. This results in a more easily maintainable codebase that can reach a wide variety of consumers.

The Disadvantage of manipulating devices like they are files comes from a decrease in efficiency. If a developer struggles to make a good driver for one of their devices, they will not be able to get the full functionality out of that device.

## **2 Would it be possible for the user to develop a new command interpreter using the system call interface provide by the operating system? How?**

It is possible for the user to develop command interpreter using the system-call interface provided by the operating system. This is possible because of how the abstraction of a system is handled. A system-call interface is meant to give a developer full control into the tasks being performed by the system devices. This includes using any part of the system to provide input, perform a process, and provide an output. This means a developer can use the system-call commands to create command interpreter of their own. ' [1]

## References

- [1] Gagne Silberschatz, Galvin. *Operating System Concepts eighth edition*. John Wiley Sons, 2009.