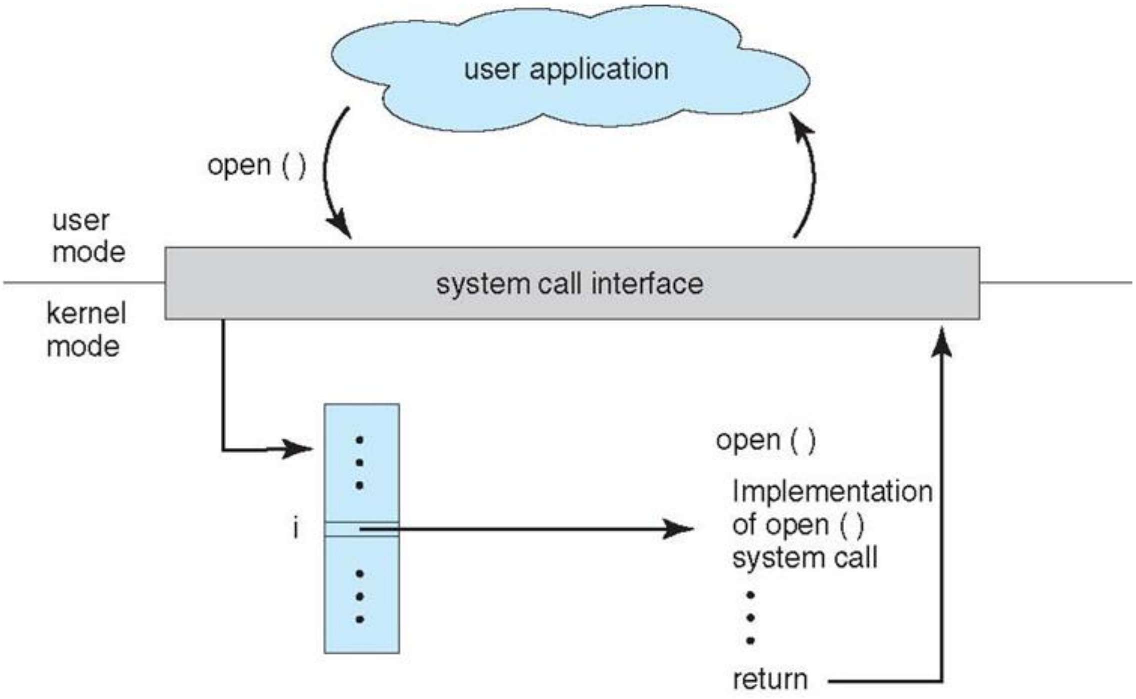

Source: https://i.stack.imgur.com/tGPV0.png

# 1st Mandatory Assignment

*Operating Systems(62588) Lecture 3*

**Teacher:** Bhupjit Singh

**To be handed in as group on DTU Learn**

## Overview & Purpose

Get familiar with features of OS, the interface exposed via system calls and learn by trying yourself

## Features to learn

1. How CLI (Shell) works
2. System Calls

## Tasks

1) Read about system calls here and here.

2) Read about the following system calls in Linux man pages(you will need to use them for your assignment). There is a very nice description with example programs on how to use pipe, wait, fork and other system calls at the end of the man pages.

   a) **Fork, Wait, Exec, Exit, Pipe, Dup2.**

3) Design a simple command line shell on the specified UNIX platform.

   a) You can take inspiration from shell program provided in Fig 1.19 of the text. You can also find example code for fork, exec and pipe in on DTU learn under Mandatory Assignment Shell. The program in figure 1.19 of the text is not complete. You must include parts of the code that are missing such that it can compile

   b) The program should work as a shell (CLI) which can receive commands with arguments and then execute them, and after completing the task return to the shell prompt waiting for next command.

4) Finally, add pipe feature to the program where the output from the first program will become the input to the second program.

## Report Requirements

1. Write a simple manual describing how to use the shell program you created. The manual should contain enough detail for a beginner to UNIX to use it. For example, you

should explain the concepts of system calls, I/O redirection, the program environment, and background program execution. The manual MUST be named: readme and must be a simple text document capable of being read by a standard Text Editor.

2. The source code MUST be commented extensively. Please structure the code well so that to allow your peers to understand. Properly commented and laid out code is much easier to interpret, and it is in your interests to ensure that peer while marking your project is able to understand your coding without having to perform mental gymnastics!
3. You MUST include bibliography for the assignment.
4. You should submit the zip file on learn.inside.dtu.dk under assignments.
5. The submission should contain only source code file(s), include file(s), and the readme file (all lowercase, please), bibliography. No executable program should be included. The group marking your project will be automatically rebuilding your shell program from the source code provided.
6. If the submitted code does not compile, it cannot be marked!
7. For instance, the files in the submitted directory would be:
   myshell.h
   myshell.c
   readme
   bibliography

   MAKE SURE YOU DOCUMENT THE SOURCES, FOR THE CODE AND OTHER ASPECTS OF REPORT.
   UNDOCUMENTED COPIED TEXT AND CODE SNIPPETS ARE LIABLE TO PLAGIARISM.

# Peer review rubric

### *General Comments*

1) The feedback should be constructive.
2) It should reflect your understanding of the assignment
3) Make sure that code snippets and text is well documents are sources are provided.

### *Feedback criteria:*

1) Does the program fulfill the requirements of the assignment?
2) Is the shell program able to receive commands with arguments
3) Does the shell program use pipes.
4) Does the shell program use input redirection.
5) Does the shell use the system variable $PATH.

6) How well is the program explained?

*It should also include your reflections of the program*

1) What have your learnt from the peers assignment
2) What would you have done differently?