Linux System Calls Cheat Sheet

1. Process Control

- fork(): Create a new child process.
- exec(): Replace current process image with new program.
- wait(): Wait for a child process to finish.
- exit(): Terminate the process.
- getpid(): Get current process ID.
- getppid(): Get parent process ID.
- kill(): Send signal to a process (e.g., terminate).
- sleep(): Pause execution for a given time.

2. File Management

- open(): Open a file and return file descriptor.
- read(): Read data from file into buffer.
- write(): Write data from buffer to file.
- close(): Close opened file descriptor.
- lseek(): Move the file pointer (read/write position).
- creat(): Create a new file.
- stat(): Get file metadata (size, owner, etc.).
- chmod(): Change file permissions.
- chown(): Change file ownership.

3. Directory Management

- mkdir(): Create a new directory.
- rmdir(): Remove an empty directory.

5. Interprocess Communication (IPC)

- pipe(): Create a unidirectional communication channel between processes.
- dup(): Duplicate a file descriptor.

Extra: Useful for Viva

- - System calls transfer control from user mode to kernel mode.
- - Invoked using software interrupts (e.g., int 0x80 or syscall instruction).
- - Every system call has a unique ID number.
- - Use `strace ./a.out` to see the list of system calls a program uses.