Assignment7 Report 1

1.1 Implementation Details

Why the Child Process Does Not Have a Controlling Terminal:

When a new session is created by the child process using setsid(), the child becomes the session leader of this new session. As part of this, the child process also loses its controlling terminal if it had one. This is because sessions are designed to be independent entities in terms of controlling terminals, allowing for background processes and daemons that do not need user interaction. PID, PGRP, and TPGID Values:

PID (Process ID): Unique identifier for each process. PGRP (Process Group ID): ID of the process group. A process group is a collection of one or more processes, usually associated with the same job, that can receive signals collectively. TPGID (Terminal Process Group ID): ID of the foreground process group of the controlling terminal. If a process does not have a controlling terminal, this will be -1. In the case of our child process:

The PID will be the unique identifier for the child process. The PGRP will be the same as the child's PID since it becomes a process group leader. The TPGID will be -1, indicating no controlling terminal. Compile and run this program in a Unix-like environment to observe the behavior. Make sure to have necessary permissions and configurations to execute system calls like fork() and setsid().