AROR UNIVERSITY OF ART, ARCHITECTURE, DESIGN & HERITAGE SUKKUR

Operating System
Lab-07.2 (LAB Task)

Compiled by: Aurangzeb Magsi

Course Title: OperatingSystem
Lab-7.2
LAB Task on Process in LINUX

Understanding Process in LINUX

Task 1: Create Multiple Child Processes

Objective: Practice using `fork()` to create multiple child processes.

Write a C program where the parent creates **two child processes** using `fork()`.
Each child should print its PID and the parent's PID.
The parent should print its PID and wait for both children to finish before exiting.

Instructions:

- Write a C program where the parent process creates **two child processes**.
- Each child process should:
  - Print its own Process ID (PID).
  - Print its Parent Process ID (PPID).
- The parent process should:
  - Print its PID.
  - Use `wait()` system call to wait for both child processes to terminate before it exits.

Hints:

- Use `fork()` twice to create two children.
- Use `wait()` twice to ensure the parent waits for both children.

Task 2: Process Hierarchy and Sleep Timing

Objective: Explore process execution timing with `sleep()`.

Write a C program where:

- The parent forks a child process.
- The child process prints its PID and then sleeps for **10 seconds** before printing "Child is awake."
- Meanwhile, the parent prints "Parent waiting for child to wake up..." and uses `wait()` to wait for the child.

Observe how the parent is blocked until the child finishes.

Instructions:

- Write a C program where:
  - The parent process creates one child process using `fork()`.
  - The child process:
    - Prints its PID.
    - Sleeps for **10 seconds** using `sleep(10)`.
    - After waking up, prints "Child is awake."
  - The parent process:

- Prints "Parent waiting for child to wake up..."
- Uses `wait()` to wait until the child finishes.

Expected Behavior:

- The parent should remain blocked until the child process completes its execution.

Task 3: ps Command and Process Inspection

Objective: Use Linux commands to inspect processes.

Perform the following:

1. Open a terminal.
2. Run a program that creates a long-sleeping process (e.g., write a program that sleeps for 30 seconds).
3. While the process is sleeping, use the following commands and capture screenshots: `o ps`
    - `ps -e`
    - `ps -ef`
    - `pidof your_program`

Instructions:

1. Open a Linux terminal.
2. Write and run a C program that creates a sleeping process (e.g., sleep for 30 seconds).
3. While the program is running and the process is sleeping:
    - Open a new terminal window.
    - Run and capture (screenshot) the output of the following commands:
        - `ps`
        - `ps -e`
        - `ps -ef`
        - `pidof your_program`

Submission Instructions

- Submit your C code files (.c).
- Submit screenshots where required.
- Include a short text file or document explaining your observations if asked.