**Processes in UNIX**

**LAB # 03**



**Fall 2024**

**CSE-302L**

**Systems Programming Lab**

Submitted by: **AIMAL KHAN.**

Registration No.: **21PWCSE2014**

Class Section: **A**

“On my honor, as student of University of Engineering and Technology, I have neither given nor received unauthorized assistance on this academic work.”

Student Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_

Submitted to:

**Engr. Abdullah Hamid**

Department of Computer Systems Engineering

University of Engineering and Technology, Peshawar

**CSE 302L: SYSTEMS PROGRAMMING LAB**

**LAB ASSESSMENT RUBRICS**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Criteria & Point Assigned** | **Outstanding 2** | **Acceptable 1.5** | **Considerable 1** | **Below Expectations 0.5** | **Score** |
| **Attendance and**  **Attentiveness in Lab**  PLO08 | Attended in proper Time and attentive in Lab | Attended in proper Time but not attentive in Lab | Attended late but attentive in Lab | Attended late not attentive in Lab |  |
| **Capability of writing Program/Algorithm/Drawing Flow Chart**  PLO1, PLO2, PLO3, PLO5 | Right attempt/ no errors and well formatted | Right attempt/ no errors but not well formatted | Right attempt/ minor errors and not well formatted | Wrong attempt |  |
| **Result or Output/ Completion of target in Lab**  PLO9 | 100% target has been completed and well formatted. | 75% target has been completed and well formatted. | 50% target has been completed but not well formatted. | None of the outputs are correct. |  |
| **Overall, Knowledge**  PLO10, | Demonstrates excellent knowledge of lab | Demonstrates good knowledge of lab | Has partial idea about the Lab and procedure followed | Has poor idea about the Lab and procedure followed |  |
| **Attention to Lab Report**  PLO4, | Submission of Lab Report in Proper Time i.e., in next day of lab, with proper documentation. | Submission of Lab Report in proper time but not with proper documentation. | Late Submission with proper documentation. | Late Submission very poor documentation |  |

**Instructor:**

|  |  |
| --- | --- |
| Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
|  |  |

**Processes in UNIX**

Objectives:

In this lab we will learn about the creation of fork, wait function call.

Tasks:

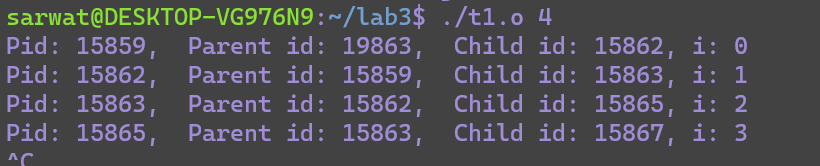
**Task 1**: Create process chain as shown in figure 3.1(b) and fill the figure 3.1 (b) with actual IDs. The program shall take a single command-line argument that specifies the number of processes to be created. Before exiting, each process shall output its i value (loop variable), its process ID (using getpid()), its parent process ID (getppid()) and the process ID of its child (return value of fork). The parent does not execute wait.

**Code:**

A screenshot of a computer

Description automatically generated

**Output:**



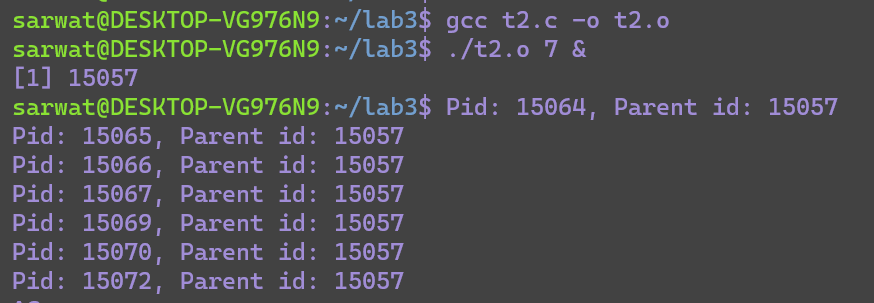
**Task 2**: Create process fan as shown in figure 3.1 (a) and fill the figure 3.1 (a) with actual IDs.

**Code:**

A screenshot of a computer program

Description automatically generated

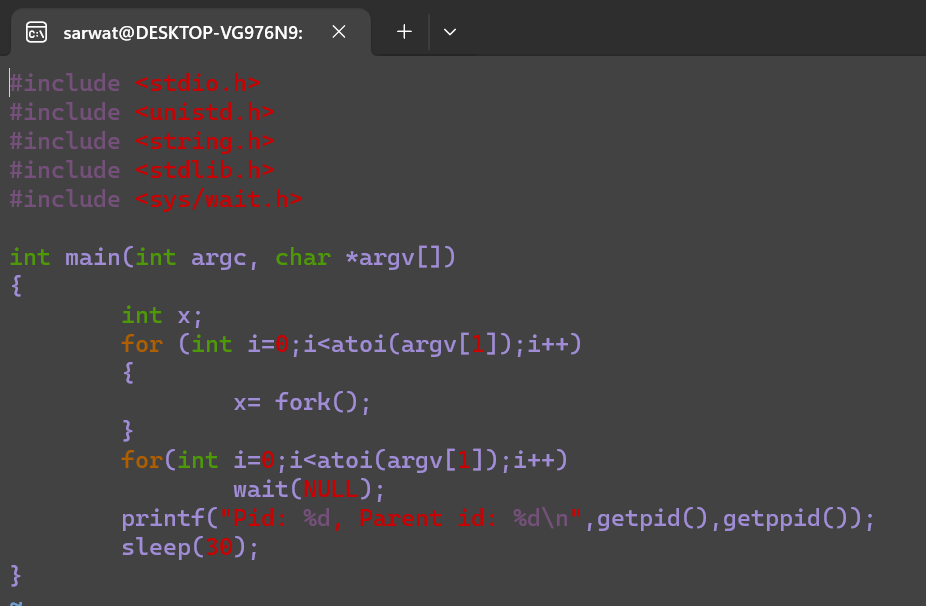
**Output:**



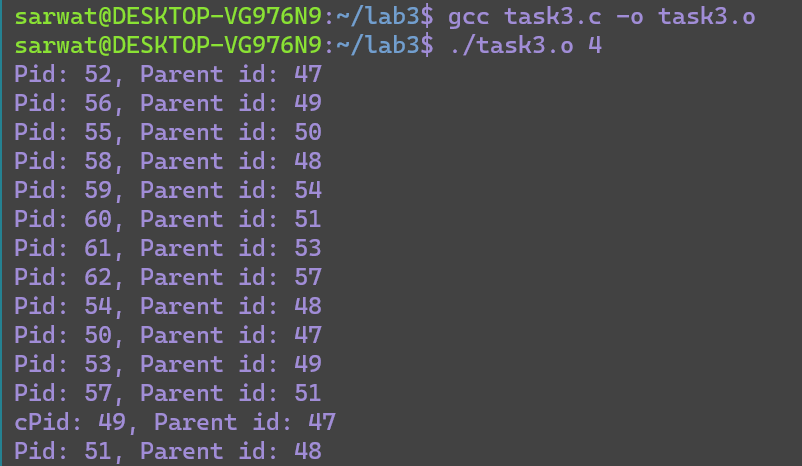
**Task3:**

Create process tree as shown in figure 3.2 and fill figure 3.2 with actual IDs.

**Code:**



**Output:**



**Task5:**

Write a program that takes N number of integers as argument and displays the factorials of N integers (print 1 only if integers are not less than zero, 0 or 1). Create separate child process for each integer. Make sure no child is orphan/zombie.

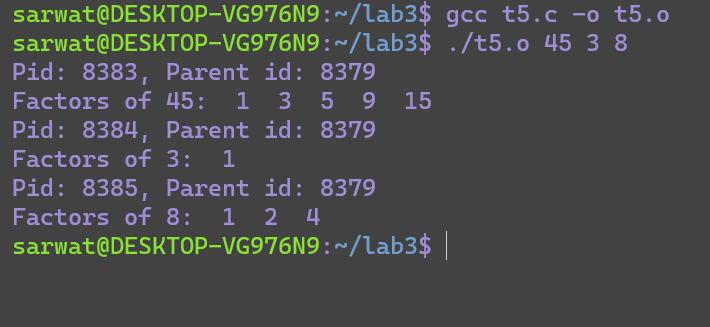
**Code:**

A screenshot of a computer program

Description automatically generated



**Code:**



Conclusion:

In conclusion, I have learned about how to creation fork ,and wait function call.