

Session: 2021 – 2025

**Submitted by:**

Abdul Mateen 2021-CS-190

**Supervised to:**

Dr. Abqa Javed

Department of Computer Science

**University of Engineering and Technology**

**Lahore Pakistan**

# **Description:**

The Given programs calculates the Factorial using IPC (PIPE Implementation). The program is written in C Language.

# **Explanation**

## **Headers**

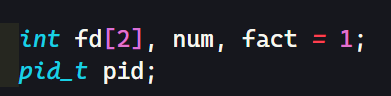
The header used are



* The **stdio.h** header file allows us to perform input and output operations in C.
* The **stdlib.h** header file allows us to declares various utility functions for type conversions, memory allocation and algorithms.
* The **unistd.h** provides access to the POSIX operating system API.

## **Variables**

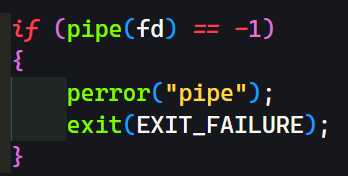
After that we declared some variables



* **Fd** is an array that will hold the file descriptors for the read and write ends of the pipe.
* **Num** will hold the number entered by the user.
* **Fact** will hold the factorial of that number.
* **pid** is a variable of type **pid\_t**, which will hold the process ID of the child process created by fork().

## **Pipe Creation**

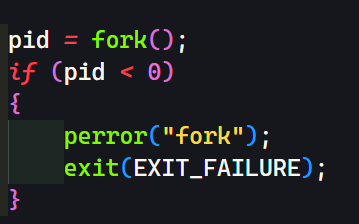
Next, a pipe is created



**pipe()** creates a pipe and returns two file descriptors: **fd[0]** is the read end of the pipe, and fd[1] is the write end of the pipe. If **pipe()** returns -1, an error has occurred and the program will exit.

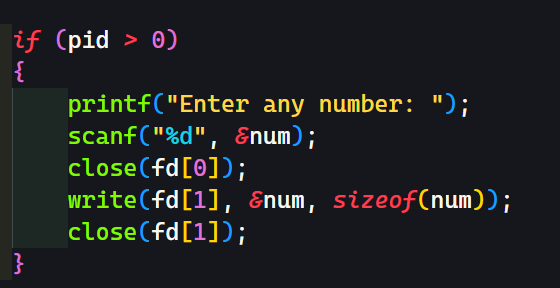
## **Child Process**

Next a child process is created



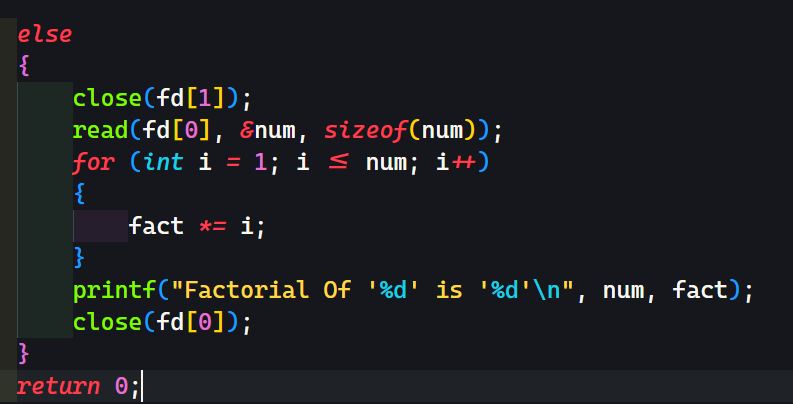
**fork()** creates a new child process by duplicating the calling process. If **fork()** returns -1, an error has occurred and the program will exit.

**If fork()** returns a positive value, the parent process is executing. The parent process prompts the user to enter a number, reads the number, and writes it to the pipe:



**close(fd[0])** closes the read end of the pipe, since the parent process will be writing to the pipe. **write(fd[1], &num, sizeof(num))** writes the number entered by the user to the write end of the pipe. **close(fd[1])** closes the write end of the pipe, since the parent process has finished writing to the pipe.

If **fork()** returns 0, the child process is executing. The child process reads the number from the pipe, calculates its factorial, and prints it to the console:



# **Code**

This is entire code of the Solution



# **Output**

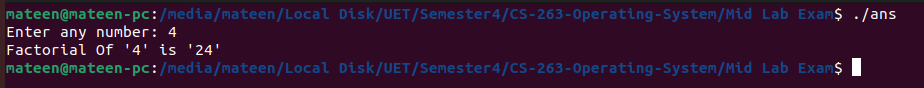
This code takes input from user as it starts

E:\UET\Semester4\CS-263-Operating-System\Mid Lab Exam\1.png

Takes input

E:\UET\Semester4\CS-263-Operating-System\Mid Lab Exam\2.png

Perform calculations



And gives output on terminal.