**UIT2512---Operating Systems Practices Lab**

# System Calls Programming in Python

# Name: Vasundhara.B

# Roll no: 3122 21 5002 119

**Aim:**

To write a python code to create 3-child processes from same parent process using fork ()

PARENT PROCESS 🡪 input for a number ‘n' and followed by printing its PID and its PPID.

CHILD1 🡪 printing its PID, PPID and check whether the given number ‘n’ is odd or even.

CHILD2 🡪 prints its PID, PPID and computes the sum of digits of ‘n’.

CHILD 🡪 prints its PID, PPID, and checks whether the given number ‘n’ is a palindrome or not.

**CODE:**

import os,time

def p1(n):

  if n%2==0:

    print("EVEN")

  else:

    print("ODD")

def p2(n):

  s=0

  n=str(n)

  for i in n:

    s = s + int(i)

  print(s)

def p3(n):

  s=str(n)

  if s==s[::-1]:

    print("Palindrome")

  else:

    print("Not palindrome")

if \_\_name\_\_=="\_\_main\_\_":

  n=int(input("Enter the number: "))

  pid = os.fork()

  if (pid != 0) :

    time.sleep(2)

    p1(n)

    print(f"child[1] --> pid = {os.getpid()} and ppid = {os.getppid()}\n")

  else:

    pid1 = os.fork()

    if (pid1 != 0):

       time.sleep(2)

       p2(n)

       print(f"child[2] --> pid = {os.getpid()} and ppid = {os.getppid()}\n")

    else:

       pid2 = os.fork()

       if (pid2 != 0) :

          time.sleep(2)

          p3(n)

          print(f"child[3] --> pid = {os.getpid()} and ppid = {os.getppid()}\n")

       else:

          time.sleep(5)

          print(f"parent --> pid = {os.getpid()}\n")

**OUTPUT:**

1. **When ‘n’ is ODD**

**A screenshot of a computer

Description automatically generated**

1. **When ‘n’ is EVEN**

A screenshot of a computer

Description automatically generated