UIT2512---Operating Systems Practices Lab

2) Creating a Zombie Process using Python

Name: Vasundhara.B

Roll no: 3122 21 5002 119

Here, you have to write a program in Python to create One Parent - One Child, where the Parent process terminates before the child completes its execution. The child returns the exit status to the parent, but the parent is not existing and causing the entry for the child process even after it becomes as the dead process.

CODE:

```python
import os
import time

def fibonacci(n):
    if n <= 0:
        return []
    elif n == 1:
        return [0]
    elif n == 2:
        return [0, 1]
    else:
        fib_series = [0, 1]
        for i in range(2, n):
            next_term = fib_series[-1] + fib_series[-2]
            fib_series.append(next_term)
        return fib_series

# Create a child process
cpid = os.fork()

if cpid == 0:
    # Print child's PID and parent's PID
    print(f"\nChild: I have the PID: {os.getpid()}, my parent's PID:
{os.getppid()}")


    fib_series = fibonacci(10)
    print(f"\nChild: Generated Fibonacci series: {fib_series}")
```

```python
# Code for parent
else:
    print(f"\nParent: I have the PID: {os.getpid()}, my child's PID: {cpid}")
    print("\nParent: I am going to sleep!!")

    # Make the parent process sleep for 4 seconds
    time.sleep(4)

    print("\nParent: I have awoken!!")

    # Execute the 'ps' command with the '-l' flag
    os.execlp("ps", "ps", "-l")
```

OUTPUT:

```
Parent: I have the PID: 8189, my child's PID: 8190

Parent: I am going to sleep!!

Child: I have the PID: 8190, my parent's PID: 8189

Child: Generated Fibonacci series: [0, 1, 1, 2, 3, 5, 8, 13, 21, 34]

Parent: I have awoken!!
PID   USER     TIME  COMMAND
   1 root      0:42 node server_8080.js
1992 repl      0:01 [javac]
2531 repl      0:01 [javac]
8109 repl      0:00 python3 main.py
8181 repl      0:00 python3 centraldepedidoautomatizada2.py
8189 repl      0:00 ps -l
8190 repl      0:00 [python3]
8192 repl      0:00 bash -c javac Main.java && java Main
8193 repl      0:01 javac Main.java
```

"ps -l" command:

This command lists processes along with detailed information