

Ex. No.: 5
Date: 19.02.2025

SYSTEM CALLS PROGRAMMING

Aim:

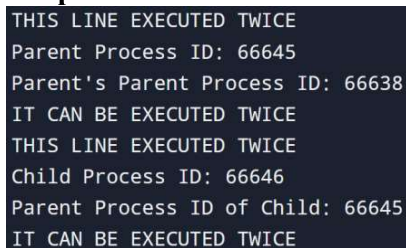
To experiment system calls using fork(), execlp() and pid() functions.

Program:

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>

int main() {
    int pid;
    pid = fork();
    if (pid == -1) {
        printf("CHILD PROCESS NOT CREATED\n");
        exit(0);
    }
    printf("THIS LINE EXECUTED TWICE\n");
    if (pid == 0) {
        printf("Child Process ID: %d\n", getpid());
        printf("Parent Process ID of Child: %d\n", getppid());
    }
    else {
        printf("Parent Process ID: %d\n", getpid());
        printf("Parent's Parent Process ID: %d\n", getppid());
    }
    printf("IT CAN BE EXECUTED TWICE\n");
    return 0;
}
```

Output:



```
THIS LINE EXECUTED TWICE
Parent Process ID: 66645
Parent's Parent Process ID: 66638
IT CAN BE EXECUTED TWICE
THIS LINE EXECUTED TWICE
Child Process ID: 66646
Parent Process ID of Child: 66645
IT CAN BE EXECUTED TWICE
```

Result:

The system calls fork(), getpid(), and getppid() were successfully used to create a child process, print process details, and show that both parent and child execute the same code.