LAB 2: EXPLORING SYSTEM CALLS

SOURCE CODE:

```
#include <stdio.h>
#include <unistd.h>
#include <fcntl.h>
#include <sys/types.h>
#include <sys/stat.h>

int main() {
    pid_t pid;
    char *filename = "example.txt";
    mode_t mode = S_IRUSR | S_IWUSR | S_IRGRP | S_IROTH; // File permissions: 644

    // Fork the process
    pid = fork();

    if (pid < 0) {
        // Error occurred
        perror("Fork failed");
        return 1;
    } else if (pid == 0) {
        // Child process
        int fd = open(filename, O_WRONLY | O_CREAT, mode);
        if (fd == -1) {
            perror("Error opening/creating file in child process");
            return 1;
        }
        close(fd);
        printf("Child process: File '%s' created successfully. PID: %d\n", filename, getpid());
    } else {
```

```
        // Parent process
        printf("Parent process: Child PID = %d, Parent PID = %d\n", pid, getpid());
    }

    return 0;
}
```

OUTPUT:

```
PS C:\Users\aksha\OneDrive - SSN Trust\Desktop\AKSHARA\COLLEGE\SEM 3\OS LAB> gcc lab2.c
PS C:\Users\aksha\OneDrive - SSN Trust\Desktop\AKSHARA\COLLEGE\SEM 3\OS LAB> ./a.exe
Parent process: Child PID = 4567, Parent PID = 4566
Child process: File 'example.txt' created successfully. PID: 4567
PS C:\Users\aksha\OneDrive - SSN Trust\Desktop\AKSHARA\COLLEGE\SEM 3\OS LAB>
```