```c
#include <stdio.h>

#include <unistd.h>

#include <stdlib.h>

#include <sys/wait.h>

int main() {
    pid_t pid;

    printf("Parent process started. PID: %d\n", getpid());

    pid = fork();  // Create child process

    if (pid < 0) {
        // Fork failed
        perror("Fork failed");
        exit(1);
    } else if (pid == 0) {
        // Child process
        printf("Child process created. PID: %d\n", getpid());

        // Replace child process with "ls -l" command
        execl("/bin/ls", "ls", "-l", NULL);

        // If exec fails
        perror("exec failed");
        exit(1);
    } else {
```

```c
        // Parent process
        printf("Parent is waiting for child to finish...\n");
        wait(NULL);
        printf("Child finished. Parent resumes.\n");
    }

    return 0;
}
```