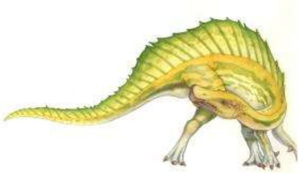
CODE 1/3

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
int main(){
    // Fork a child process
    pid_t child_pid = fork();
    int status;
    if (child_pid == -1) {
        // Fork failed
        printf("Fork failed");
        exit(EXIT_FAILURE);
    }
```



CODE 2/3

```
if (child_pid > 0) {
    // Parent process
    printf("Parent process (PID: %d)\n", getpid());
    printf("Child process created with PID: %d\n", child_pid);
    wait(&status);
    printf("Child exited with status: %d\n", status);
}

else if (child_pid == 0) {
    // Child process
    printf("Child process (PID: %d)\n", getpid());
    exit(EXIT_FAILURE);
    //exit(EXIT_SUCCESS);
}
```

CODE 3/3

```
    // Code executed by both parent and child processes
    printf("This code is executed by both parent and child processes.\n");

    return 0;
}
```