1st part

- Firstly, I have created the fork() command.
- As we don't know which process(child or parent) will execute first; I have created an if statement, in which if the child process is executed first, it will wait for the parent process to execute.
- In other if-else statement, the code block of the parent process will start executing, and print Statement A.
- Then, the parent process will wait for the child process; then, the child process will execute from line no. 20. After the complete execution of the child process, the parent process executes.
- Conclusion: During the child process, fork() returns zero; while during the parent process, fork() returns the PID of the child process.

2nd part

- Firstly, I have created two functions of void type, which print the Factorial of the given number and the Fibonacci Series till a given term.
- Using the fork() command, we have to call the Factorial function in the child process and the Fibonacci Series function in the parent process.
- The order of the code should be such that the parent process should wait for the child process to execute
- Hence, to solve the problem, if the parent process is executed first, then I have made an if statement in which the parent process has to wait for the child process to be executed.
- Conclusion: First Factorial function will be called and then Fibonacci Series function will be called.

2nd part (bonus)

- Firstly, I have created two functions of void type, which print the Factorial of the given number and the Fibonacci Series till a given term.
- Using the fork() command, we have to call the Factorial function in the child process and the Fibonacci Series function in the parent process.
- The order of the code should be such that the parent process should wait for the child process to execute
- Hence, to solve the problem, if the child process is executed first, then I have made an if statement in which the child process has to wait for the parent process to be executed.
- Conclusion: First Fibonacci Series function will be called and then Factorial function will be called.