

Task 02

1 Problem 1 [4 marks]

- a) Write a C program that creates a child process. **After** creating the child process, the parent process should output two messages; first, it should print "**I am the parent process**", then it should print "**The parent is done**". Similarly, the child process should print "**I am the child process**" and "**The child is done**".
- b) Make sure, the second message of the child is always done **before** the second message of the parent.

2 Problem 2 [4 marks]

- a) Repeat the above exercise 1a) using Pthreads. Adjust the messages accordingly.
- b) Make sure, the second message of the spawned thread is always done **before** the second message of the main thread.
- c) Change your implementation, so that the main thread prints his second message **before** the spawned thread does.

3 Problem 3 [1 mark]

Modify your solution for Problem 1 such that the parent process always prints its second message **before** that of the child process. Make sure that all prints still happen **after** the child process has been created!

4 Problem 4 [1 mark]

Devise a program that allows you to explore what happens when a program that has more than one thread is creating new processes. Find out, whether the behaviour depends on which thread is creating new processes or on how many threads create new processes. Summarize your findings in 200-300 words.