Lab 06

OS Lab tasks:

Task 1

You have to write a program that will display 2^6 times “Hello” using at least 4 fork () calls. Each child will must print Hello at least once. With the program, you have to create the parent child relationship tree of this program.

Task 2

Create a file called **catcount.c** in which you will write a program that receives one command line argument of type *string*: the name of a text file. Your program will work as follows:

- Start out by spawning a child process.
- The child process calls **exec** to run with the command line argument that the parent received. Read the man page of this library call to learn what arguments to pass to it. This replaces the binary executable on the child process, but the parent goes on with the code you wrote.
- The parent process calls **wait** so that it blocks until the child terminates and passes back its termination status.
- If the child process terminates without error, the parent spawns another child and, again, calls **wait** so that it can block until the child terminates.
- The new child calls **exec** again, but this time it runs on the same argument that the parent received from the command line (the file name passed to **cat** previously).
- Once the parent learns that the child has terminated, it goes on to terminate also. If the parent gets to this point, it’s because all has gone well, so its termination status should be 0