Homework 2 – Processes in UNIX

Write a C program for UNIX that creates several processes, according to the following scenario:

- The initial process will be called the parent process (P)
- The parent process (P) creates two child processes, let's call them Child1 and Child2
- The first child process (Child1) creates another process, GrandChild1
- The second child process (Child2) creates another process, GrandChild2

More precisely, the following list describes the behavior of each process:

1. Process P will create process Child1, then will immediately create process Child2 and then will wait for both child processes to finish their execution; in the end it will display the message "I am the Parent process and my pid is … . Both my children finished their execution." (the ellipsis stand for the actual pid value which should be an integer number)

2. Process Child1 will create process GrandChild1, will wait for process GrandChild1 to finish its execution, and then will display the message "I am the process Child1 and my pid is … ." (the ellipsis stand for the actual pid value which should be an integer number)

3. Process Child2 will create process GrandChild2, will wait for process GrandChild2 to finish its execution, and then will display the message "I am the process Child2 and my pid is … ." (the ellipsis stand for the actual pid value which should be an integer number)

4. Process GrandChild1 will display the message "I am the process GrandChild1 and my pid is … . My parent is process Child1 and it has the following pid: … ." (the ellipsis stand for actual pid values which should be integer numbers)

5. Process GrandChild1 will display the message "I am the process GrandChild1 and my pid is … . My parent is process Child1 and it has the following pid: … ." (the ellipsis stand for actual pid values which should be integer numbers)

HINTS:
- The "fork()" system call is used to create a new process
- The "fork()" system call may return an error code, which should be detected and reported
- The "wait()" system call is used when waiting for a child process to finish is execution
- Work on your computer or online using https://replit.com

DELIVERABLES:
- The source code for the C program, stored in the text file "hw2.c"
- A screenshot showing the results of your program's execution, stored in a graphic file like "hw2.png" or "hw2.gif" or "hw2.pdf", etc.