Course	60-256	October 21, 2018
Instructor	Dr. B. Boufama	
Assignment	03, Due on October 31, 11.59pm	

The pseudo-code below, behaves like a simple shell for interpreting simple bash scripts, that consist of a few lines. Each line is made of a single shell command with or without arguments.

Write a C program that uses only Unix input/outputs system calls to open the script file and to read each line until the end of the file. Each line consists of a command, except lines that start with **#** or that are empty, that should be ignored. Your program uses fork() and exec() to execute the commands.

Synopsis: **myShell** < **myScriptFile** >

```
Pseudo-code:
------------
  Open Scriptfile                    // use perror() in case of error
  while(1){
    readLine from myScriptFile
    if (read-error or end-of-file)  // use perror() in case of error
      quit
    if (line is a comment or empty)
      skip iteration (continue in C)

    extract command and arguments from line and store them in
    a NULL-terminated array of strings, call it myCommand

    duplicate current process
    parent process waits for its child to terminate
    child should exec to the new program using myCommand
  }
```

Hints:

- You can reuse **int readLine**(), from your assignment 02, to read lines from files.
- You can use **sscanf()** to extract the strings from a line, read by readLine(), and store them in array myCommand (myCommand is an array of strings).

Here is an example of text file myScriptFile:

```
date
# date was printed, this line is a comment and should be skipped

ls /bin
grep -i bigFoot /etc/bigFootArchive
cp letter.txt letterOld.txt
rm letter.txt
clear
#end of script file
```