B1 – Shell Programming

B-PSU-110

minishell1

Bootstrap

minishell1

binary name:	my_exec
repository name:	PSU_my_exec_$ACADEMICYEAR
repository rights:	ramassage-tek
language:	C
compilation:	via Makefile, including re, clean and fclean rules

- Your repository must contain the totality of your source files, but no useless files (binary, temp files, obj files,…).

- Error messages have to be written on the error output, and the program should then exit with the 84 error code (0 if there is no error).

Write a program that takes one string as parameter, that contains a program name with its path and arguments.
Your program must execute the program with these arguments, the following way:

```
~/B-PSU-110> ./my_exec "/bin/ls -l /dev"
Program name:  /bin/ls
Nb args:  2
PID: 1346
Child PID: 1348
...  .  .
...  .  .
...  .  .   /* execution of ls -l /dev */
...  .  .
...  .  .
Program terminated.
Status:  OK
```

```
~/B-PSU-110> ./my_exec "./my_segfault"
Program name:  ./my_segfault
Nb args:  0
PID: 1513
Child PID: 1514
...  .  .
...  .  .
...  .  .   /* execution of ./my_segfault */
...  .  .
...  .  .
Program terminated.
Status:  Segmentation fault
```

Authorized Functions

- malloc, free, exit, opendir, readdir, closedir, getcwd, chdir
- fork, stat, lstat, fstat, getcwd, open, close
- read, write, execve, access, isatty, wait, waitpid
- wait3, wait4, signal, kill, getpid, strerror, perror, strsignal