Test Cases

|  |
| --- |
| Process creation and execution - foreground   * test1a: ps↵ ps f↵ ls -l -t -r↵ ps -e -a -f -u↵ * test1b: /bin/ps f↵ * test1c: ls -l -a↵ wrong↵ ps f↵ man pipe↵ ps f↵ * test1d: just press "enter" or space a few times * test1e: ./forever↵ \* ps f↵   \* Use another terminal to send the SIGKILL signal to the ./forever process |
| Process creation and execution – background + SIGCHLD signal   * test2a: ps & ls↵ ps & ls & * test2b: ./loopf 4&↵ ps f↵ * test2c: ./loopf 4 &↵ ps f↵ * test2d: ./loopf 15&↵ ./loopf 8&↵ ps f↵ 🕤 ps f↵ * test2e: ps f &↵ ps f &↵ ps f &↵ ps f↵ * test2f: ./loopf 20 &↵ ./loopf 9 &↵ ps f↵ ./loopf 10↵ * test2g: ./loopf 20&↵ ./loopf 20 &↵ ./forever↵🕤 \* ps f↵   🕤 Wait for the background processes to terminate first  \* Use another terminal to send the SIGKILL signal to the ./forever process |
| Process creation and execution – use of ‘|’   * Test3a: ls |↵ * Test3b: ls -l| wc -l↵ ls -l| wc -l↵ * Test3c: cat PASS2.txt | grep myshell | wc -l↵ ps f↵ * Test3d: ./inproc | ./outproc | ./outproc | ./outproc | ./outproc↵ ps f↵ * Test3e: ./loopf 15 | ./loopf 6 | ./loopf 10↵ ps f↵ * Test3f: cat loopever.c | wrong | wc -l↵ ps f↵ |
| Support the exit built-in command with the defined behavior   * Test4a: exit↵ * Test4b: exit anyway↵ * Test4c: not exit↵ |
| Support the timeX built-in command with the defined behavior   * Test5a: timeX↵ ./timeX↵ timeX ./loopf 5 &↵ * Test5b: timeX ps f↵ timeX ls -l -t -r↵ * Test5c: timeX cat PASS2.txt | grep myshell | wc -l↵ * Test5d: timeX ./loopf 9↵ * Test5e: timeX ./loopf 15 | ./loopf 6 | ./loopf 10↵ |
| Support creation of multiple pipelined processes, which are running in background together with a foreground job, with the defined behavior   * Test6a: ./loopf 20 | ./loopf 9 | ./loopf 12&↵ ps f↵ 🕤 ps f↵ * Test6b: ./loopf 20 | ./loopf 9 | ./loopf 12&↵ ./loopf 15↵ 🕤 ps f↵   🕤 Wait for the processes to terminate first |
| Handle SIGINT signal   * Test7a: Ctrl-c 🕇 Ctrl-c 🕇 Ctrl-c 🕇 * Test7b: ./loopf 10↵ Ctrl-c # Ctrl-c 🕇 ps f↵ * Test7c: ./forever↵ Ctrl-c # Ctrl-c # Ctrl-c # \* ps f↵ * Test7d: ./loopf 20&↵ ./loopf 25&↵ Ctrl-c 🕇 ps f↵ * Test7e: ./loopf 20&↵ ./loopf 25&↵ ./loopf 5↵ Ctrl-c 🕇 ps f↵   # Use Ctrl-c to interrupt current running process  🕇 Use Ctrl-c to interrupt microshell process  \* Use another terminal to send the SIGKILL signal to the ./forever process |
| Support the viewtree built-in command with the defined behavior   * Test8a: viewtree↵ * Test8b: ./viewtree↵ viewtree something↵ * Test8c: ./loopf 30&↵ ./loopf 25&↵ viewtree↵ * Test8d: ./nested 4 &↵ ./loopf 30&↵ ./loopf25&↵ viewtree * Test8e: ./myshell↵ viewtree↵ * Test8f: ./myshell↵ ./loopf 30&↵ ./loopf 25&↵ viewtree↵ |
|  |