Hermmy Wang

704978214

hermmyw@hotmail.com

**Lab 1 Report**

1. Test cases

|  |  |  |
| --- | --- | --- |
| Bash-4.4 | Dash | Simple shell command |
| *Test case 1:*  *Delete every x, y, and z from pg98.txt and sleep 1 second* | | |
| $ sleep 1 | cat pg98.txt | tr -d "xyz" >1bout.txt 2> 1berr.txt; times | $ sleep 1 | cat pg98.txt | tr -d "xyz" >1dout.txt 2> 1derr.txt; times | ./simpsh \  --profile \  --creat --rdwr io.txt \  --rdonly pg98.txt \  --creat --wronly 1sout.txt \  --creat --wronly 1serr.txt \  --pipe \  --pipe \  --command 0 0 0 sleep 1 \  --command 1 7 3 cat \  --command 6 2 3 tr -d "xyz" \  --close 5 \  --close 7 \  --wait |
| *Test case 2:*  *Replace the string "we" with "shoes" in pg98.txt, then count lines* | | |
| $ cat pg98.txt | sed 's/we/shoes/g' | wc -l >2bout.txt 2> 2berr.txt; times | $ cat pg98.txt | sed 's/we/shoes/g' | wc -l >2dout.txt 2> 2derr.txt; times | ./simpsh \  --profile \  --rdonly pg98.txt \  --pipe \  --pipe \  --creat --wronly 2sout.txt \  --creat --wronly 2serr.txt \  --command 0 2 6 cat \  --command 1 4 6 sed 's/we/shoes/g' \  --command 3 5 6 wc -l \  --close 2 \  --close 4 \  --wait |
| *Test case 3:*  *replace a, b, c with x, y, z, sort pg98.txt, ignoring leading blanks and case, and count the number of the string "spy"* | | |
| $ tr "abc" "xyz" <pg98.txt | sort -bf | grep -ic "spy" >3bout.txt 2>3berr.txt; times | $ tr "abc" "xyz" <pg98.txt | sort -bf | grep -ic "spy" >3dout.txt 2>3derr.txt; times | ./simpsh \  --profile \  --rdonly pg98.txt \  --pipe \  --pipe \  --creat --wronly 3sout.txt \  --creat --wronly 3serr.txt \  --command 0 2 6 tr "abc" "xyz" \  --command 1 4 6 sort -bf \  --command 3 5 6 grep -ic "spy" \  --close 2 \  --close 4 \  --wait |

2. Data

Each 3 test cases are run on Bash-4.4, Dash, and simpsh three times. The result is obtained by taking the average of the three trials. The table data shows at most three decimal spaces after rounding for each entry. Bold red font represents the slowest performance in each time category of each test case, and bold green represents the best. If a row does not have a bold green font, there is a tie.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test case** | **Time(s)** | **Bash** | **Dash** | **Simpsh** | **Best** |
| 1 | User | **0.001** | 0.000 | 0.000 | Dash/Simpsh |
| User\_child | **0.023** | 0.007 | **0.002** | Dash |
| System | **0.012** | 0.000 | 0.000 | Dash/Simpsh |
| System\_child | **0.041** | 0.013 | **0.004** | Simpsh |
| 2 | User | **0.002** | 0.000 | 0.000 | Dash/Simpsh |
| User\_child | **0.022** | 0.020 | **0.011** | Simpsh |
| System | **0.005** | 0.000 | 0.000 | Dash/Simpsh |
| System\_child | **0.012** | **0.003** | 0.004 | Dash |
| 3 | User | **0.002** | 0.000 | 0.000 | Dash/Simpsh |
| User\_child | **0.228** | 0.230 | **0.073** | Simpsh |
| System | **0.006** | 0.000 | 0.000 | Dash/Simpsh |
| System\_child | **0.023** | 0.013 | **0.007** | Simpsh |

For each time category in the 3 benchmarks, Bash all have the worst CPU time.

For test case #1, Dash and Simpsh perform equally well. Simpsh is faster on child process system time, while Dash is better at user time. Parent user time and system time are all approximately zero.

For test case #2, Dash and Simpsh again perform equally well. This time, Simpsh is faster on child process user time, while Dash is better at system time. Parent user time and system time are all approximately zero.

For test case #3, which is a more complex benchmark involving the most commands, Simpsh has the best performance in every time category. Especially in child process user and system CPU time, Simpsh is more than twice faster than Dash.

In conclusion, Bash appears to be the slowest shell among the three. It shows the slowest performance in all 12 data entries. Simpsh outperforms the other shells, especially in child process system time. Out of the 12 data entries, Simpsh has the best performance in 5, and ties with Dash in 6. Therefore, Simpsh is the fastest shell.