Lab 1C Report

I created 3 benchmarks, each with Following commands:

|  |  |  |
| --- | --- | --- |
| **Benchmark** | **Shell Type** | **Command** |
| **1** | bash | cat < /usr/share/dict/words 2> err1.txt | sort -r 2> err1.txt | grep and > bout1.txt 2> err1.txt |
| dash | cat < /usr/share/dict/words 2> err1.txt | sort -r 2> err1.txt | grep and > dout1.txt 2> err1.txt |
| simpsh | ./simpsh --rdonly /usr/share/dict/words --wronly err1.txt --pipe --pipe --wronly out1.txt --command 0 3 1 cat --command 2 5 1 sort -r --command 4 6 1 grep and --wait --profile |
| **2** | bash | cat < /usr/share/dict/words 2> err2.txt | tr "a" "b" 2> err2.txt | sort 2> err2.txt | grep b > bout2.txt 2> err2.txt |
| dash | cat < /usr/share/dict/words 2> err2.txt | tr "a" "b" 2> err2.txt | sort 2> err2.txt | grep b > dout2.txt 2> err2.txt |
| simpsh | ./simpsh --rdonly /usr/share/dict/words --wronly out2.txt --wronly err2.txt --pipe --pipe --pipe --command 0 4 2 cat --command 3 6 2 tr "a" "b" --command 5 8 2 sort --command 7 1 2 grep b --wait --profile |
| **3** | bash | man -k file < /dev/null 2> err3.txt | tr -s ' ' '\n' 2> err3.txt | sort 2> err3.txt | sort -r 2> err3.txt | grep il 2> err3.txt | tr '\n' ' ' 2> err3.txt | wc -c > bout3.txt 2> err3.txt |
| dash | man -k file < /dev/null 2> err3.txt | tr -s ' ' '\n' 2> err3.txt | sort 2> err3.txt | sort -r 2> err3.txt | grep il 2> err3.txt | tr '\n' ' ' 2> err3.txt | wc -c > dout3.txt 2> err3.txt |
| simpsh | ./simpsh --rdonly /dev/null --wronly out3.txt --wronly err3.txt --pipe --pipe --pipe --pipe --pipe --pipe --command 0 4 2 man -k file --command 3 6 2 tr -s ' ' '\n' --command 5 8 2 sort --command 7 10 2 sort -r --command 9 12 2 grep il --command 11 14 2 tr '\n' ' ' --command 13 1 2 wc -c --wait --profile |

Below is the chart of user/system times for 3 benchmarks:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Benchmark 1** | | | | | | | | | |
| Trial | bash | | | dash | | | simpsh | | |
| 1 |  | User | System |  | User | System |  | User | System |
| Parent | 0.002 | 0 | Parent | 0 | 0 | Parent | 0 | 0.001361 |
| Child | 1.118 | 0.019 | Child | 1.15 | 0.01 | Child | 1.157977 | 0.019058 |
| Total | 1.12 | 0.019 | Total | 1.15 | 0.01 | Total | 1.157977 | 0.020419 |
| 2 |  | User | System |  | User | System |  | User | System |
| Parent | 0.001 | 0.002 | Parent | 0 | 0 | Parent | 0 | 0.00138 |
| Child | 1.14 | 0.019 | Child | 1.13 | 0.02 | Child | 1.133302 | 0.023737 |
| Total | 1.141 | 0.021 | Total | 1.13 | 0.02 | Total | 1.133302 | 0.025117 |
| 3 |  | User | System |  | User | System |  | User | System |
| Parent | 0 | 0.002 | Parent | 0 | 0 | Parent | 0.001501 | 0 |
| Child | 1.126 | 0.021 | Child | 1.12 | 0.02 | Child | 1.139985 | 0.014913 |
| Total | 1.126 | 0.023 | Total | 1.12 | 0.02 | Total | 1.141486 | 0.014913 |
|  |  | User | System |  | User | System |  | User | System |
| **Average** |  | **1.129** | **0.021** |  | **1.13333** | **0.01667** |  | **1.14426** | **0.02015** |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Benchmark 2** | | | | | | | | | | |
| Trial | bash | | | dash | | | simpsh | | | |
| 1 |  | User | System |  | User | System |  | User | System |
| Parent | 0.001 | 0.002 | Parent | 0 | 0 | Parent | 0 | 0.001398 |
| Child | 1.158 | 0.026 | Child | 1.15 | 0.02 | Child | 1.152663 | 0.028553 |
| Total | 1.159 | 0.028 | Total | 1.15 | 0.02 | Total | 1.152663 | 0.029951 |
| 2 |  | User | System |  | User | System |  | User | System |
| Parent | 0.001 | 0.002 | Parent | 0 | 0 | Parent | 0 | 0.001555 |
| Child | 1.162 | 0.036 | Child | 1.17 | 0.02 | Child | 1.161075 | 0.025047 |
| Total | 1.163 | 0.038 | Total | 1.17 | 0.02 | Total | 1.161075 | 0.026602 |
| 3 |  | User | System |  | User | System |  | User | System |
| Parent | 0.001 | 0.001 | Parent | 0 | 0 | Parent | 0 | 0.001438 |
| Child | 1.154 | 0.023 | Child | 1.16 | 0.02 | Child | 1.164126 | 0.026854 |
| Total | 1.155 | 0.024 | Total | 1.16 | 0.02 | Total | 1.164126 | 0.028292 |
|  |  | User | System |  | User | System |  | User | System |
| **Average** |  | **1.159** | **0.03** |  | **1.16** | **0.02** |  | **1.15929** | **0.02828** |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Benchmark 3** | | | | | | | | | |
| Trial | bash | | | dash | | | simpsh | | |
| 1 |  | User | System |  | User | System |  | User | System |
| Parent | 0.001 | 0.001 | Parent | 0 | 0 | Parent | 0.003118 | 0.012474 |
| Child | 0.165 | 0.013 | Child | 0.16 | 0.01 | Child | 0.165785 | 0.018391 |
| Total | 0.166 | 0.014 | Total | 0.16 | 0.01 | Total | 0.168903 | 0.030865 |
| 2 |  | User | System |  | User | System |  | User | System |
| Parent | 0 | 0.002 | Parent | 0 | 0 | Parent | 0.001555 | 0.003111 |
| Child | 0.166 | 0.011 | Child | 0.15 | 0.01 | Child | 0.170475 | 0.024623 |
| Total | 0.166 | 0.013 | Total | 0.15 | 0.01 | Total | 0.17203 | 0.027734 |
| 3 |  | User | System |  | User | System |  | User | System |
| Parent | 0.001 | 0.002 | Parent | 0 | 0 | Parent | 0.000616 | 0.001805 |
| Child | 0.159 | 0.018 | Child | 0.15 | 0.01 | Child | 0.168751 | 0.016411 |
| Total | 0.16 | 0.02 | Total | 0.15 | 0.01 | Total | 0.169367 | 0.018216 |
|  |  | User | System |  | User | System |  | User | System |
| **Average** |  | **0.164** | **0.01567** |  | **0.15333** | **0.01** |  | **0.1701** | **0.02561** |

Conclusion

Comparing the user times, there seems to be no noticeable difference, although bash seems to be faster in terms of user times overall.

Comparingthe system times, this dash seems to lead the other two, although no noticeable difference exists either.

In focusing the simpsh user time, it’s either between bash/dash, or is slower than both. This is probably due to the creation of initial variables, and extra checking of command grammars, etc (a lot of validity checks are done within simpsh).

In terms of system time, it’s also between bash/dash, or is slower than both. This is probably due to extra allocation of pid array, file descriptor array, and all the extra output done by wait option.