CS 111 — Lab 1

Benchmark testing

Sarah Le Cam - February 6, 2017
UCLA ID: 804-099-507

Benchmark 1

This benchmark is taken from the command prompt. Seeing as it is a good example of how the program should be used, I thought that it would be appropriate to include it in my testing.

./simpsh

```
./simpsh \
 --profile \
 --rdonly a1.txt \
 --pipe \
 --pipe \
 --creat --trunc --wronly c \
 --creat --append --wronly d \
 --command 3 5 6 tr A-Z a-z \
 --command 0 2 6 sort \
 --command 1 4 6 cat b - \
 --wait > test1out.txt
```

bash/dash

```
( (sort < a1.txt | cat b - | tr A-Z a-z > c) 2>>d)
```

Benchmark 2

./simpsh

```
./simpsh \
  --profile \
  --rdonly a1.txt \
  --pipe \
  --pipe \
  --creat --trunc --wronly c \
  --creat --append --wronly d \
  --command 0 2 6 uniq -D \
  --command 1 4 6 sort \
  --command 3 5 6 cat -b \
  --wait > test2out.txt
```

bash/dash

```
( (uniq -D < a1.txt | sort | cat -b > c) 2>>d)
```

Benchmark 3

./simpsh

```
./simpsh \
  --profile \
  --rdonly a1.txt \
  --pipe \
  --pipe \
  --pipe \
  --creat --trunc --wronly c \
  --creat --append --wronly d \
  --command 0 2 8 uniq -u \
  --command 1 4 8 sort \
  --command 3 6 8 wc -w \
  --command 5 7 8 cat \
  --wait > test3out.txt
```

bash/dash

```
( (uniq -u < a1.txt | sort | wc -w | cat > c) 2>>d)
```

Results

All of the test were run with the provided a1.txt as input. Provided times are averages over 4 runs.

	./simpsh	bash	dash
Benchmark 1	Parent Processes: - User: 0.000120 seconds - System: 0.001746 seconds Children Processes: - User: 6.848384 seconds - System: 0.188947 seconds	Parent Processes: - User: 0.002 seconds - System: 0.000 seconds Children Processes: - User: 6.785 seconds - System: 0.183 seconds	Parent Processes: - User: 0.000000 seconds - System: 0.000000 seconds Children Processes: - User: 8.620000 seconds - System: 0.090000 seconds
Benchmark 2	Parent Processes: - User: 0.000342 seconds - System: 0.002485 seconds Children Processes: - User: 1.093136 seconds - System: 0.021559 seconds	Parent Processes: - User: 0.002 seconds - System: 0.001 seconds Children Processes: - User: 2.235 seconds - System: 0.024 seconds	Parent Processes: - User: 0.000000 seconds - System: 0.000000 seconds Children Processes: - User: 1.260000 seconds - System: 0.070000 seconds
Benchmark 3	Parent Processes: - User: 0.000000 seconds - System: 0.002552 seconds Children Processes: - User: 5.014513 seconds - System: 0.062991 seconds	Parent Processes: - User: 0.000 seconds - System: 0.002 seconds Children Processes: - User: 6.070 seconds - System: 0.070 seconds	Parent Processes: - User: 0.000000 seconds - System: 0.000000 seconds Children Processes: - User: 6.460000 seconds - System: 0.210000 seconds

Conclusions

In general, my implementation of simpsh seems to run in approximately the same time as the corresponding commands in bash. There is a slight difference in terms of the run time of the run times of the children processes in tests 2 and 3, for which my simpsh requires less time. The dash parent processes consistently require 0 seconds for both user and system time. However, the children processes require more time than when run with bash or simpsh. In conclusion, my simpsh shell seems to be the most efficient by a small margin.