UID: 905116878

Summary:

		User: Own	Sys: Own	User: Child	Sys: Child
Test 1:	Simpsh	211 usec	1723 usec	1073 usec	4285 usec
	Bash	1600 usec	1400 usec	1800 usec	4800 usec
	Dash	1000 usec	2000 usec	2400 usec	4400 usec
Test 2:	Simpsh	230 usec	1507 usec	1144 usec	2587 usec
	Bash	1600 usec	2000 usec	1200 usec	4000 usec
	Dash	1000 usec	2200 usec	1000 usec	4600 usec
Test 3:	Simpsh	930 usec	762 usec	811 usec	3584 usec
	Bash	1200 usec	1800 usec	1400 usec	4600 usec
	Dash	1000 usec	1400 usec	1800 usec	3800 usec

Simpsh:

(1) `./simpsh --rdonly input.txt --creat --rdwr test1out.txt --creat --wronly test1err.txt --pipe --pipe --creat --wronly translate.txt --command 0 4 2 sed 's/ //g' --command 3 6 2 sort --command 5 7 2 tr [A-Z] [a-z] --close 4 --close 5 --close 6 --wait`

(2) `./simpsh --profile --rdonly input.txt --creat --append --rdwr test2out.txt --creat --excl --wronly test2err.txt --pipe --command 0 1 2 sleep 1 --command 0 4 2 cat --command 3 1 2 wc --close 4 --wait`

(3) `./simpsh --pipe --rdonly input.txt --creat --wronly out.txt --creat --wronly err.txt --pipe --command 2 1 4 tr " " "\n" --command 0 6 4 head -n 20 --command 5 3 4 grep -c n --close 0 --close 1 --close 6 --wait`

Bash/Dash:

(1) `sed 's/ //g' <input.txt 2> test1err.txt | sort 2>test1err.txt | tr [A-Z] [a-z] >translate.txt 2> test1err.txt`

(2) `sleep 1 < input.txt > test1out.txt 2> test1err.txt`
`cat <input.txt 2> test1err.txt | wc 2> test1err.txt >test1out.txt`

(3) `tr " " "\n" < input.txt 2>err.txt | head -n 20 2> err.txt |grep -c n 2> err.txt`

Conclusion:
Based off of the experimental data, my implementation of simpsh was overall faster in both user and system time for the shell itself, as well as the processes called by the shell. The differences were most drastic in the user time of the shell itself with a little more mixed findings in the other aspects of measurement.

Overall bash and dash times were generally the same with dash being slightly faster overall for the user time of the shell, but mixed for all other aspects of measurement.

Trials run to calculate averages (optional):
SIMPSH:

Test 1 ./simpsh --rdonly input.txt --creat --rdwr test1out.txt --creat --wronly test1err.txt --pipe --pipe --creat --wronly translate.txt --command 0 4 2 sed 's/ //g' --command 3 6 2 sort --command 5 7 2 tr [A-Z] [a-z] --close 4 --close 5 --close 6 --wait

	User: Own	Sys: Own	User: Child	Sys: Child
1	0 usec	2113 usec	567 usec	5130 usec
2	0 usec	1887 usec	1044 usec	4532 usec
3	1055 usec	698 usec	799 usec	4043 usec
4	0 usec	1906 usec	1373 usec	4222 usec
5	0 usec	2013 usec	1583 usec	3499 usec
Average:	211 usec	1723 usec	1073 usec	4285 usec

Test 2: ./simpsh --profile --rdonly input.txt --creat --append --rdwr test2out.txt --creat --excl --wronly test2err.txt --pipe --command 0 1 2 sleep 1 --command 0 4 2 cat --command 3 1 2 wc --close 4 --wait

	User: Own	Sys: Own	User: Child	Sys: Child
1	0 usec	1847 usec	1418 usec	2246 usec
2	0 usec	1688 usec	1744 usec	1904 usec
3	1149 usec	527 usec	1086 usec	2770 usec

4	0 usec	1694 usec	1474 usec	2392 usec
5	0 usec	1780 usec	0 usec	3621 usec
Average:	230 usec	1507 usec	1144 usec	2587 usec

Test case 3:
```
./simpsh --pipe --rdonly input.txt --creat --wronly out.txt --creat
--wronly err.txt --pipe --command 2 1 4 tr " " "\n" --command 0 6 4
head -n 20 --command 5 3 4 grep -c n --close 0 --close 1 --close 6
--wait
```

	User: Own	Sys: Own	User: Child	Sys: Child
1	1713 usec	0 usec	1275 usec	3602 usec
2	942 usec	709 usec	735 usec	3579 usec
3	978 usec	722 usec	493 usec	4111 usec
4	1018 usec	748 usec	548 usec	3775 usec
5	0 usec	1631 usec	1006 usec	2851 usec
Average:	930 usec	762 usec	811 usec	3584 usec

BASH:
```
#test 1
sed 's/ //g' <input.txt 2> test1err.txt | sort 2>test1err.txt | tr
[A-Z] [a-z] >translate.txt 2> test1err.txt
```

	User: Own	Sys: Own	User: Child	Sys: Child
1	.001s	.002s	.003s	.004s
2	.003s	.000s	.002s	.004s
3	.001s	.002s	.001s	.006s
4	.002s	.001s	.002s	.004s
5	.001s	.002s	.001s	.006s
Average:	1600 usec	1400 usec	1800 usec	4800usec

```
#test 2
sleep 1 < input.txt > test1out.txt 2> test1err.txt
cat <input.txt 2> test1err.txt | wc 2> test1err.txt >test1out.txt
```

	User: Own	Sys: Own	User: Child	Sys: Child
1	.002s	.002s	.000s	.005s
2	.000s	.003s	.002s	.003s
3	.002s	.002s	.001s	.004s
4	.002s	.001s	.002s	.004s
5	.002s	.002s	.001s	.004s
Average:	1600 usec	2000 usec	1200 usec	4000 usec

```
#test3
tr " " "\n" < input.txt 2>err.txt | head -n 20 2> err.txt |grep -c n
2> err.txt
```

	User: Own	Sys: Own	User: Child	Sys: Child
1	.000s	.003s	.001s	.006s
2	.003s	.000s	.000s	.005s
3	.003s	.000s	.002s	.004s
4	.000s	.003s	.003s	.003s
5	.000s	.003s	.001s	.005s
Average:	1200 usec	1800 usec	1400 usec	4600 usec

DASH:

```
#test 1
sed 's/ //g' <input.txt 2> test1err.txt | sort 2>test1err.txt | tr
[A-Z] [a-z] >translate.txt 2> test1err.txt
```

	User: Own	Sys: Own	User: Child	Sys: Child
1	.001s	.002s	.004s	.004s
2	.001s	.001s	.001s	.005s
3	.001s	.002s	.001s	.006s
4	.001s	.003s	.002s	.004s
5	.001s	.002s	.004s	.003s
Average:	1000 usec	2000 usec	2400 usec	4400 usec

```
#test 2
sleep 1 < input.txt > test1out.txt 2> test1err.txt
cat <input.txt 2> test1err.txt | wc 2> test1err.txt >test1out.txt
```

	User: Own	Sys: Own	User: Child	Sys: Child
1	.001s	.002s	.003s	.005s
2	.002s	.001s	.000s	.005s
3	.000s	.003s	.001s	.004s
4	.000s	.003s	.001s	.004s
5	.002s	.002s	.000s	.005s
Average:	1000 usec	2200 usec	1000 usec	4600 usec

```
#test3
tr " " "\n" < input.txt 2>err.txt | head -n 20 2> err.txt |grep -c n
2> err.txt
```

	User: Own	Sys: Own	User: Child	Sys: Child
1	.000s	.003s	.001s	.006s
2	.001s	.001s	.002s	.003s
3	.002s	.001s	.002s	.003s

4	.001s	.001s	.002s	.004s
5	.001s	.001s	.002s	.003s
Average:	1000 usec	1400 usec	1800 usec	3800 usec