

## CS 4513 Project 1

## Dumpster Diving

## Design

I created a folder named *experiment\_folder* on the */tmp* partition. Within this folder, I created 10 folders named *folder1*, *folder2*....*folder10*. Each of these folders contained 10 files with the *.txt* extension – thus, the file names were *file1*, *file2*, ...*file10*. Each folder contained copies of these 10 files. Each file contained portions of the novel “Alice’s Adventures in Wonderland” by Lewis Carroll. The text of the novel was obtained from Project Gutenberg.<sup>1</sup>

The shell script I used to test the times measured the elapsed time using the *date* shell command - *start\_time`date +%N`*. Using the *%N* specifier allows for the measurement of time in nanoseconds. After executing the *rm* program, the script measured the time again - *end\_time`date +%N`*. The time taken by the program was obtained by the difference - *echo execution time was `expr \$end\_time - \$start\_time` nanoseconds*.

I performed ten runs – one on each folder – and calculated the throughput. I recorded my data by using the *du* utility to measure the size of the file or directory being transferred, and by noting the time as described above. The throughput was obtained by dividing the size by the time.

I performed this experiment on a laptop that was connected to the WPI Wireless network. I used the PuTTY software to connect to the CCC server (where I had developed my code), and to run the experiments using the shell. I was also running the Google Chrome browser, Microsoft Powerpoint and Word, and WinSCP at the time on the laptop. The Microsoft Skydrive sync

---

<sup>1</sup> The full text can be obtained at <http://www.gutenberg.org/cache/epub/11/pg11.txt>.

service was also running in the background, which may have contributed to a delay in the connection to the CCC server.

I also wrote a simple C file that renamed *argv[1]* to *argv[2]*, *argv[3]* to *argv[4]*, etc. I made 10 *rename* requests in the file, and measured the time of execution. I calculated the average time for those ten requests and repeated with the same set of files 10 times.

## Results

The preliminary run of the script was to copy just a single folder to the trash directory. The trash directory was in my user folder, thus ensuring that it was on a different partition as compared to the */tmp* partition. A custom trash directory was specified using the *-t* flag - *./rm\_prog -t ~/cs4513/proj1/trash\_dir/ -r -i /tmp/experiment\_folder/folder5/*.

This program took 98852000 nanoseconds to execute, which is 98.85 milliseconds. The size of the folder was:

```
$du trash_dir/folder5/ -b
164798 trash_dir/folder5/
```

Thus, the throughput was  $1.6672 \times 10^6$  bytes per second (Bps). The full set of results are as in the table below-

Table 1: Results for *rm -r* on directories

Filename	Size (bytes)	Time (milliseconds)	Throughput ( $10^6$ Bps)
<i>folder1</i>	164798	80.17	2.055606835
<i>folder2</i>	164798	50.7	3.250453649
<i>folder3</i>	164798	59.54	2.767853544
<i>folder4</i>	164798	177.4	9.289627959

<i>folder5</i>	164798	98.85	1.667152251
<i>folder6</i>	164798	108.4	1.520276753
<i>folder7</i>	164798	160.1	1.02934416
<i>folder8</i>	164798	812.7	2.02778393
<i>folder9</i>	164798	447.6	3.68181412
<i>folder10</i>	164798	141.8	1.162186178
<b>Mean</b>	<b>164798</b>	<b>213.726</b>	1.495279597
<b>Standard Deviation</b>	<b>0</b>	<b>239.285</b>	9.794490255

The results for the *rename* operation are presented below:

Table 2: data on the rename operation

Run	Total execution time	Average time for 10 requests
	(nanoseconds)	(seconds)
1	466602000	0.0466602
2	82282000	0.0082282
3	45542000	0.0045542
4	604152000	0.0604152
5	436599000	0.0436599
6	50995000	0.0050995
7	779484000	0.0779484
8	264973000	0.0264973
9	636366000	0.0636366
10	154521000	0.0154521

<b>Mean</b>	<b>352151600</b>	<b>0.03521516</b>
<b>Standard Deviation</b>	<b>269014161.6</b>	<b>0.026901416</b>

### Analysis

There is great variation observed in the results for the *rm* utility. Since the size of the folders removed to trash were the same, I hypothesize that the variation in time is due to both network delays and possible delays between caching and flushing to disk.

The data on the *rename* operation shows less standard deviation, thereby showing that the average time of 0.035 seconds per rename operation can be used as a good approximation.