

Write Up Assignment 3

William Kudsk

CRUZID: wkudsk

CSE 130, Fall 2019

Testing:

I used curl to test, using get, put with content length, and as well as some requests that would fail

like incorrect filenames and forbidden or non-existent files. I used two main files to test with:

one was a file with Hello World, the other was the entire script of Macbeth, written three times.

I used a test script with these commands:

```
curl -T tests/SmallFile0123456XYZxyz-mmmm http://localhost:8080 --request-target
ABCDEFabcdef012345XYZxyz-mm -v
curl http://localhost:8080 --request-target ABCDEFabcdef012345XYZxyz-mm -v > t1
curl http://localhost:8080 --request-target ABCDEFabcdef012345XYZxyz-mm -v
curl http://localhost:8080 --request-target out1 -v
curl http://localhost:8080 --request-target CBCDEFabcdef012345XYZxyz-mm -v
curl http://localhost:8080 --request-target BBCDEFabcdef012345XYZxyz-mm -v
curl http://localhost:8080 --request-target BBCDEFabcdef012345XYZxyz-mm -v > out1
curl http://localhost:8080 --request-target jabberwockypoem1234567890xy -v > jabberwocktest
curl -T tests/SmallFile0123456XYZxyz-mmmm http://localhost:8080 --request-target
SmallFile0123456XYZxyz-mmmm -v
curl http://localhost:8080 --request-target SmallFile0123456XYZxyz-mmmm -v
cat out1 | diff - tests/BigFile012345XYZxyz-xyzmmmm
curl -T tests/BigFile012345XYZxyz-xyzmmmm http://localhost:8080 --request-target
BBCDEFabcdef012345XYZxyz-mm -v
curl -T tests/BigFile012345XYZxyz-xyzmmmm http://localhost:8080 --request-target
BigFile012345XYZxyz-xyzmmmm -v
curl http://localhost:8080 --request-target BigFile012345XYZxyz-xyzmmmm -v
cat BBCDEFabcdef012345XYZxyz-mm | diff - tests/BigFile012345XYZxyz-xyzmmmm
```

smallFile: 1KB

BigFile: 98KB

Jabberwocky: 4234 KB

Questions:

For the script above:

With Caching:

real 0m5.006s

user 0m0.133s

sys 0m0.119s

Without Caching:

real 0m4.846s

user 0m0.119s

sys 0m0.103s

After doing several runs: With caching varies heavily each time because 4.8 - 5.1 while without caching each run is pretty consistently around 4.8.

I expect it might be because I am not using the most efficient way of caching as I am doing a simple FIFO method. In addition, I am using the deque and string data structures from c++. If I used arrays with efficient FIFO calls to them I could possibly increase my speed.