Report: Homework 1 - Webcrawler

> My code can do HTTP 1.1 Chunked downloads

Cole McAnelly
CS 463

Introduction

Architecture

My webcrawler codebase was designed to be as extensible as possible from the very beginning of the project in part 1. I knew that I wanted to use some OOP paterns, so after creating the given `Socket` class (I renamed `Tcp`), I also created `Request` and `Response` classes inside of an `Http` namespace (`Http::Request`, `Http::Response`). The `Request` class would be passed into the `Tcp` class' request method, `Tcp::request()`, to send the request to the server, and the reponse method, `Tcp::response()`, would return a pointer to the `Http::Reponse` object it created.

This simple design allowed for me to scale this pattern from a single request all the way to a multithreaded webcrawler.

After getting to part 2, I created a `Crawler` class that would handle all of the organization and management of resources, which was very valuable in part 3. The crawler class creates a temporary `MMapFile` object, which is a windows memory mapped file for quick reading. This `MMapFile` class provides modern, ergonomic, C++ iterators that iterate over each line in the file, allowing the user to use these iterators to create multiple SLT data structures. The `Crawler` class now handles all of the atomic variables for data tracking, and the Mutex's for protected member access to as many threads as the user requests. It also holds the handles to the threads themselves, initializing and joining both the stats thread and the many crawler threads.

Additionally, I had some smaller classes that were more meant to aid the program, and not be integral. The most important of these is my own custom `Mutex<T>` class, which provides ergonimic handling of C++ `std::mutex`'s by associating them with the data itself. The `::lock()` method returns a gaurded pointer to the underlying data, to ensure that the data cannot be accessed in 2 places at once (providing synchronization). The `Iter` class is a simple forward iterater that returns an optional value, finally returning `std::nullopt` when the iterator has reached the end of its iterable.

The `WinSock` class simple provides RAII access to the windows socket API by automatically calling the library linking functions upon creating and dropping.

Lessons Learned

I had never written code for networking this in-depth, nor had I ever written code on my own Windows laptop--preferring to use WSL to use Unix system calls. I learned so much specifically about Windows Developement, including Visual Studio 2022, like how useful a visual debugger is for debugging threads rather than just a TUI similar to something provided by GDB. I learned about socket developement, and learned new ways that I could use the SLT data structures and algorithms to make socket development feel more ergonomic, while not sacrificing performance.

Full Trace (1M URLs)

```
Opened C:\Users\colem\Downloads\URL-input-1M.txt with size 66152005
[  2] 1000 Q 985599 E   14405 H   2988 D   2125 I  1374 R   238 C   206 L    2K
      *** crawling 103.0 pps @ 9.0 Mbps
[  4] 1000 Q 977648 E   22356 H   4153 D   3279 I  2088 R   387 C   349 L    7K
      *** crawling 35.8 pps @ 6.0 Mbps
[  6] 1000 Q 973344 E   26660 H   4835 D   3877 I  2486 R   467 C   452 L   10K
      *** crawling 17.2 pps @ 2.6 Mbps
[  8] 1000 Q 971677 E   28327 H   5096 D   4094 I  2631 R   491 C   482 L   12K
      *** crawling 3.8 pps @ 1.3 Mbps
[ 10] 1000 Q 969927 E   30077 H   5344 D   4280 I  2763 R   513 C   508 L   13K
      *** crawling 2.6 pps @ 0.3 Mbps
[ 12] 1000 Q 967236 E   32768 H   5758 D   4631 I  3008 R   548 C   537 L   14K
      *** crawling 2.4 pps @ 0.4 Mbps
[ 14] 1000 Q 957568 E   42436 H   7083 D   5690 I  3736 R   656 C   630 L   16K
      *** crawling 6.6 pps @ 0.9 Mbps
[ 16] 1000 Q 947932 E   52072 H   8437 D   6857 I  4513 R   805 C   773 L   19K
      *** crawling 8.9 pps @ 1.3 Mbps
[ 18] 1000 Q 942364 E   57640 H   9167 D   7486 I  4958 R   889 C   866 L   23K
      *** crawling 5.2 pps @ 0.8 Mbps
[ 20] 1000 Q 938103 E   61901 H   9754 D   7988 I  5330 R   952 C   938 L   25K
      *** crawling 3.6 pps @ 0.6 Mbps
[ 22] 1000 Q 934110 E   65894 H  10376 D   8508 I  5711 R  1025 C  1004 L   26K
      *** crawling 3.0 pps @ 0.4 Mbps
[ 24] 1000 Q 929372 E   70632 H  11089 D   9092 I  6124 R  1095 C  1079 L   30K
      *** crawling 3.1 pps @ 0.5 Mbps
[ 26] 1000 Q 922107 E   77897 H  12098 D   9907 I  6675 R  1173 C  1154 L   31K
      *** crawling 2.9 pps @ 0.2 Mbps
[ 28] 1000 Q 912822 E   87182 H  13293 D  10946 I  7366 R  1282 C  1257 L   33K
      *** crawling 3.7 pps @ 0.6 Mbps
[ 30] 1000 Q 906220 E   93784 H  14209 D  11679 I  7893 R  1374 C  1360 L   37K
      *** crawling 3.4 pps @ 0.4 Mbps
[ 32] 1000 Q 900067 E   99937 H  15012 D  12403 I  8411 R  1461 C  1440 L   38K
      *** crawling 2.5 pps @ 0.4 Mbps
[ 34] 1000 Q 893994 E  106010 H  15866 D  13121 I  8924 R  1552 C  1530 L   44K
      *** crawling 2.6 pps @ 0.7 Mbps
[ 36] 1000 Q 886764 E  113240 H  16918 D  14016 I  9547 R  1660 C  1637 L   46K
      *** crawling 3.0 pps @ 0.4 Mbps
[ 38] 1000 Q 879821 E  120183 H  17924 D  14861 I 10129 R  1770 C  1740 L   49K
      *** crawling 2.7 pps @ 0.4 Mbps
[ 40] 1000 Q 873538 E  126466 H  18879 D  15673 I 10715 R  1871 C  1838 L   52K
      *** crawling 2.5 pps @ 0.4 Mbps
[ 42] 1000 Q 867499 E  132505 H  19777 D  16429 I 11234 R  1976 C  1951 L   55K
      *** crawling 2.7 pps @ 0.3 Mbps
[ 44] 1000 Q 862021 E  137983 H  20568 D  17091 I 11713 R  2052 C  2031 L   57K
      *** crawling 1.8 pps @ 0.2 Mbps
[ 46] 1000 Q 856889 E  143115 H  21400 D  17793 I 12203 R  2142 C  2118 L   59K
      *** crawling 1.9 pps @ 0.3 Mbps
[ 48] 1000 Q 852391 E  147613 H  22155 D  18434 I 12657 R  2213 C  2185 L   61K
      *** crawling 1.4 pps @ 0.5 Mbps
[ 50] 1000 Q 846985 E  153019 H  23005 D  19128 I 13139 R  2302 C  2275 L   62K
      *** crawling 1.8 pps @ 0.3 Mbps
[ 52] 1000 Q 841038 E  158966 H  23890 D  19876 I 13678 R  2410 C  2376 L   64K
      *** crawling 1.9 pps @ 0.3 Mbps
[ 54] 1000 Q 834447 E  165557 H  24785 D  20615 I 14189 R  2497 C  2468 L   66K
      *** crawling 1.7 pps @ 0.2 Mbps
[ 56] 1000 Q 827958 E  172046 H  25641 D  21320 I 14650 R  2573 C  2549 L   68K
      *** crawling 1.4 pps @ 0.2 Mbps
[ 58] 1000 Q 820724 E  179280 H  26431 D  21979 I 15118 R  2658 C  2627 L   69K
```

```
      *** crawling 1.3 pps @ 0.2 Mbps
[ 60] 1000 Q 814374 E  185630 H  27147 D  22509 I 15484 R  2731 C  2703 L   71K
      *** crawling 1.3 pps @ 0.2 Mbps
[ 62] 1000 Q 806064 E  193940 H  28226 D  23476 I 16167 R  2834 C  2792 L   72K
      *** crawling 1.4 pps @ 0.2 Mbps
[ 64] 1000 Q 798942 E  201062 H  29257 D  24366 I 16775 R  2940 C  2903 L   73K
      *** crawling 1.7 pps @ 0.2 Mbps
[ 66] 1000 Q 792707 E  207297 H  30082 D  25069 I 17241 R  3013 C  2976 L   77K
      *** crawling 1.1 pps @ 0.2 Mbps
[ 68] 1000 Q 787495 E  212509 H  30883 D  25738 I 17685 R  3092 C  3059 L   79K
      *** crawling 1.2 pps @ 0.2 Mbps
[ 70] 1000 Q 781762 E  218242 H  31631 D  26369 I 18106 R  3166 C  3132 L   80K
      *** crawling 1.0 pps @ 0.1 Mbps
[ 72] 1000 Q 775801 E  224203 H  32400 D  27001 I 18530 R  3232 C  3196 L   81K
      *** crawling 0.9 pps @ 0.1 Mbps
[ 74] 1000 Q 768410 E  231594 H  33312 D  27760 I 19048 R  3309 C  3276 L   82K
      *** crawling 1.1 pps @ 0.2 Mbps
[ 76] 1000 Q 763803 E  236201 H  34015 D  28334 I 19423 R  3362 C  3325 L   83K
      *** crawling 0.6 pps @ 0.1 Mbps
[ 78] 1000 Q 757652 E  242352 H  34981 D  29147 I 19983 R  3449 C  3416 L   84K
      *** crawling 1.2 pps @ 0.1 Mbps
[ 80] 1000 Q 752326 E  247678 H  35867 D  29911 I 20489 R  3546 C  3504 L   85K
      *** crawling 1.1 pps @ 0.1 Mbps
[ 82] 1000 Q 747133 E  252871 H  36771 D  30652 I 20993 R  3617 C  3581 L   88K
      *** crawling 0.9 pps @ 0.2 Mbps
[ 84] 1000 Q 741501 E  258503 H  37696 D  31460 I 21540 R  3698 C  3657 L   91K
      *** crawling 0.9 pps @ 0.2 Mbps
[ 86] 1000 Q 736247 E  263757 H  38533 D  32154 I 22016 R  3765 C  3739 L   94K
      *** crawling 1.0 pps @ 0.2 Mbps
[ 88] 1000 Q 731897 E  268107 H  39255 D  32760 I 22431 R  3840 C  3804 L   94K
      *** crawling 0.7 pps @ 0.1 Mbps
[ 90] 1000 Q 727254 E  272750 H  40082 D  33480 I 22932 R  3923 C  3886 L   97K
      *** crawling 0.9 pps @ 0.1 Mbps
[ 92] 1000 Q 723272 E  276732 H  40763 D  34006 I 23270 R  3980 C  3951 L   98K
      *** crawling 0.7 pps @ 0.1 Mbps
[ 94] 1000 Q 718116 E  281888 H  41616 D  34796 I 23776 R  4062 C  4014 L   99K
      *** crawling 0.7 pps @ 0.1 Mbps
[ 96] 1000 Q 713005 E  286999 H  42575 D  35611 I 24303 R  4143 C  4102 L  101K
      *** crawling 0.9 pps @ 0.2 Mbps
[ 98] 1000 Q 709279 E  290725 H  43234 D  36123 I 24643 R  4202 C  4161 L  103K
      *** crawling 0.6 pps @ 0.1 Mbps
[100] 1000 Q 703361 E  296643 H  44259 D  37024 I 25235 R  4311 C  4270 L  104K
      *** crawling 1.1 pps @ 0.1 Mbps
[102] 1000 Q 698672 E  301332 H  45066 D  37717 I 25672 R  4391 C  4356 L  106K
      *** crawling 0.8 pps @ 0.1 Mbps
[104] 1000 Q 694009 E  305995 H  45843 D  38390 I 26135 R  4479 C  4437 L  108K
      *** crawling 0.8 pps @ 0.1 Mbps
[106] 1000 Q 689225 E  310779 H  46657 D  39089 I 26607 R  4562 C  4525 L  110K
      *** crawling 0.8 pps @ 0.1 Mbps
[108] 1000 Q 684635 E  315369 H  47376 D  39690 I 27028 R  4631 C  4586 L  110K
      *** crawling 0.6 pps @ 0.1 Mbps
[110] 1000 Q 678628 E  321376 H  48303 D  40479 I 27523 R  4732 C  4693 L  113K
      *** crawling 1.0 pps @ 0.2 Mbps
[112] 1000 Q 673735 E  326269 H  49160 D  41198 I 28000 R  4812 C  4767 L  116K
      *** crawling 0.7 pps @ 0.1 Mbps
[114] 1000 Q 668744 E  331260 H  49959 D  41875 I 28456 R  4892 C  4849 L  119K
      *** crawling 0.7 pps @ 0.1 Mbps
[116] 1000 Q 663754 E  336250 H  50856 D  42640 I 28956 R  4982 C  4934 L  120K
      *** crawling 0.7 pps @ 0.1 Mbps
[118] 1000 Q 658977 E  341027 H  51700 D  43362 I 29414 R  5069 C  5028 L  123K
      *** crawling 0.8 pps @ 0.1 Mbps
[120] 1000 Q 654014 E  345990 H  52497 D  44024 I 29842 R  5141 C  5103 L  125K
      *** crawling 0.6 pps @ 0.1 Mbps
[122] 1000 Q 647257 E  352747 H  53476 D  44871 I 30396 R  5227 C  5180 L  127K
      *** crawling 0.6 pps @ 0.1 Mbps
[124] 1000 Q 641774 E  358230 H  54349 D  45520 I 30830 R  5299 C  5268 L  129K
      *** crawling 0.7 pps @ 0.1 Mbps
[126] 1000 Q 633086 E  366918 H  55421 D  46521 I 31524 R  5412 C  5357 L  130K
      *** crawling 0.7 pps @ 0.1 Mbps
[128] 1000 Q 624999 E  375005 H  56399 D  47329 I 32043 R  5510 C  5462 L  132K
      *** crawling 0.8 pps @ 0.1 Mbps
```

```
[130] 1000 Q 618694 E  381310 H  57244 D  48052 I 32508 R  5596 C  5544 L  134K
      *** crawling 0.6 pps @ 0.1 Mbps
[132] 1000 Q 609790 E  390214 H  58290 D  48944 I 33070 R  5708 C  5663 L  137K
      *** crawling 0.9 pps @ 0.1 Mbps
[134] 1000 Q 602634 E  397370 H  59235 D  49768 I 33605 R  5805 C  5759 L  139K
      *** crawling 0.7 pps @ 0.1 Mbps
[136] 1000 Q 594242 E  405762 H  60127 D  50528 I 34098 R  5887 C  5844 L  141K
      *** crawling 0.6 pps @ 0.1 Mbps
[138] 1000 Q 584950 E  415054 H  61218 D  51443 I 34680 R  5985 C  5940 L  142K
      *** crawling 0.7 pps @ 0.1 Mbps
[140] 1000 Q 577700 E  422304 H  62147 D  52228 I 35189 R  6084 C  6024 L  145K
      *** crawling 0.6 pps @ 0.1 Mbps
[142] 1000 Q 569826 E  430178 H  63184 D  53118 I 35730 R  6162 C  6117 L  147K
      *** crawling 0.7 pps @ 0.1 Mbps
[144] 1000 Q 563334 E  436670 H  64074 D  53885 I 36225 R  6252 C  6212 L  150K
      *** crawling 0.7 pps @ 0.1 Mbps
[146] 1000 Q 557804 E  442200 H  64855 D  54557 I 36642 R  6321 C  6277 L  152K
      *** crawling 0.4 pps @ 0.1 Mbps
[148] 1000 Q 551840 E  448164 H  65696 D  55257 I 37079 R  6387 C  6346 L  154K
      *** crawling 0.5 pps @ 0.1 Mbps
[150] 1000 Q 546507 E  453497 H  66475 D  55900 I 37488 R  6448 C  6407 L  155K
      *** crawling 0.4 pps @ 0.1 Mbps
[152] 1000 Q 540769 E  459235 H  67260 D  56548 I 37908 R  6518 C  6476 L  157K
      *** crawling 0.5 pps @ 0.1 Mbps
[154] 1000 Q 533054 E  466950 H  68223 D  57383 I 38422 R  6606 C  6564 L  160K
      *** crawling 0.6 pps @ 0.1 Mbps
[156] 1000 Q 526821 E  473183 H  69073 D  58101 I 38868 R  6686 C  6644 L  162K
      *** crawling 0.5 pps @ 0.1 Mbps
[158] 1000 Q 519778 E  480226 H  69901 D  58813 I 39304 R  6760 C  6710 L  163K
      *** crawling 0.4 pps @ 0.1 Mbps
[161] 1000 Q 511316 E  488688 H  70735 D  59512 I 39728 R  6828 C  6786 L  165K
      *** crawling 0.5 pps @ 0.1 Mbps
[163] 1000 Q 500706 E  499298 H  71680 D  60327 I 40259 R  6917 C  6867 L  166K
      *** crawling 0.5 pps @ 0.1 Mbps
[165] 1000 Q 489776 E  510228 H  72588 D  61082 I 40719 R  6999 C  6956 L  168K
      *** crawling 0.5 pps @ 0.1 Mbps
[167] 1000 Q 481044 E  518960 H  73501 D  61869 I 41197 R  7078 C  7036 L  168K
      *** crawling 0.5 pps @ 0.1 Mbps
[169] 1000 Q 473915 E  526089 H  74402 D  62634 I 41667 R  7145 C  7104 L  171K
      *** crawling 0.4 pps @ 0.1 Mbps
[171] 1000 Q 467611 E  532393 H  75202 D  63315 I 42050 R  7201 C  7163 L  172K
      *** crawling 0.3 pps @ 0.1 Mbps
[173] 1000 Q 460357 E  539647 H  76060 D  64045 I 42495 R  7276 C  7230 L  173K
      *** crawling 0.4 pps @ 0.1 Mbps
[175] 1000 Q 453590 E  546414 H  76884 D  64751 I 42916 R  7347 C  7296 L  174K
      *** crawling 0.4 pps @ 0.0 Mbps
[177] 1000 Q 447406 E  552598 H  77735 D  65485 I 43377 R  7421 C  7368 L  176K
      *** crawling 0.4 pps @ 0.1 Mbps
[179] 1000 Q 442023 E  557981 H  78549 D  66161 I 43775 R  7484 C  7437 L  182K
      *** crawling 0.4 pps @ 0.1 Mbps
[181] 1000 Q 436933 E  563071 H  79372 D  66839 I 44168 R  7551 C  7505 L  184K
      *** crawling 0.4 pps @ 0.0 Mbps
[183] 1000 Q 432555 E  567449 H  80097 D  67417 I 44488 R  7606 C  7560 L  185K
      *** crawling 0.3 pps @ 0.0 Mbps
[185] 1000 Q 428489 E  571515 H  80782 D  67978 I 44781 R  7666 C  7615 L  186K
      *** crawling 0.3 pps @ 0.1 Mbps
[187] 1000 Q 423761 E  576243 H  81501 D  68545 I 45092 R  7720 C  7673 L  188K
      *** crawling 0.3 pps @ 0.0 Mbps
[189] 1000 Q 420235 E  579769 H  82063 D  68916 I 45267 R  7755 C  7714 L  189K
      *** crawling 0.2 pps @ 0.0 Mbps
[191] 1000 Q 414806 E  585198 H  82776 D  69470 I 45548 R  7808 C  7764 L  190K
      *** crawling 0.3 pps @ 0.0 Mbps
[193] 1000 Q 406689 E  593315 H  83822 D  70406 I 46037 R  7895 C  7838 L  191K
      *** crawling 0.4 pps @ 0.0 Mbps
[195] 1000 Q 400918 E  599086 H  84622 D  71040 I 46333 R  7960 C  7911 L  193K
      *** crawling 0.4 pps @ 0.0 Mbps
[197] 1000 Q 394313 E  605691 H  85564 D  71840 I 46695 R  8026 C  7980 L  194K
      *** crawling 0.4 pps @ 0.0 Mbps
[199] 1000 Q 387390 E  612614 H  86540 D  72650 I 47025 R  8074 C  8025 L  197K
      *** crawling 0.2 pps @ 0.1 Mbps
[201] 1000 Q 380700 E  619304 H  87475 D  73471 I 47360 R  8134 C  8086 L  197K
```

```
      *** crawling 0.3 pps @ 0.1 Mbps
[203] 1000 Q 375159 E  624845 H  88258 D  74126 I 47636 R  8186 C  8140 L  201K
      *** crawling 0.3 pps @ 0.0 Mbps
[205] 1000 Q 367572 E  632432 H  89233 D  74911 I 47956 R  8237 C  8188 L  202K
      *** crawling 0.2 pps @ 0.0 Mbps
[207] 1000 Q 360508 E  639496 H  90113 D  75623 I 48258 R  8280 C  8233 L  203K
      *** crawling 0.2 pps @ 0.0 Mbps
[209] 1000 Q 354522 E  645482 H  90907 D  76273 I 48505 R  8320 C  8274 L  204K
      *** crawling 0.2 pps @ 0.0 Mbps
[211] 1000 Q 348824 E  651180 H  91720 D  76916 I 48780 R  8359 C  8316 L  205K
      *** crawling 0.2 pps @ 0.0 Mbps
[213] 1000 Q 342609 E  657395 H  92694 D  77721 I 49077 R  8410 C  8364 L  206K
      *** crawling 0.2 pps @ 0.0 Mbps
[215] 1000 Q 336666 E  663338 H  93572 D  78447 I 49359 R  8454 C  8408 L  207K
      *** crawling 0.2 pps @ 0.0 Mbps
[217] 1000 Q 330245 E  669759 H  94492 D  79208 I 49659 R  8513 C  8462 L  208K
      *** crawling 0.2 pps @ 0.0 Mbps
[219] 1000 Q 323619 E  676385 H  95442 D  79974 I 49971 R  8566 C  8514 L  210K
      *** crawling 0.2 pps @ 0.0 Mbps
[221] 1000 Q 317769 E  682235 H  96326 D  80664 I 50240 R  8603 C  8556 L  210K
      *** crawling 0.2 pps @ 0.0 Mbps
[223] 1000 Q 310398 E  689606 H  97397 D  81582 I 50611 R  8668 C  8618 L  211K
      *** crawling 0.3 pps @ 0.0 Mbps
[225] 1000 Q 304852 E  695152 H  98195 D  82231 I 50871 R  8717 C  8669 L  212K
      *** crawling 0.2 pps @ 0.1 Mbps
[227] 1000 Q 299015 E  700989 H  99077 D  82947 I 51139 R  8771 C  8722 L  212K
      *** crawling 0.2 pps @ 0.0 Mbps
[229] 1000 Q 293201 E  706803 H  99937 D  83655 I 51430 R  8828 C  8778 L  213K
      *** crawling 0.2 pps @ 0.0 Mbps
[231] 1000 Q 287354 E  712650 H 100786 D  84353 I 51689 R  8883 C  8831 L  213K
      *** crawling 0.2 pps @ 0.0 Mbps
[233] 1000 Q 281215 E  718789 H 101666 D  85083 I 51949 R  8936 C  8890 L  214K
      *** crawling 0.3 pps @ 0.0 Mbps
[235] 1000 Q 274593 E  725411 H 102544 D  85804 I 52214 R  8988 C  8935 L  215K
      *** crawling 0.2 pps @ 0.0 Mbps
[237] 1000 Q 267161 E  732843 H 103606 D  86676 I 52520 R  9025 C  8981 L  217K
      *** crawling 0.2 pps @ 0.0 Mbps
[239] 1000 Q 259454 E  740550 H 104644 D  87544 I 52834 R  9080 C  9026 L  220K
      *** crawling 0.2 pps @ 0.1 Mbps
[241] 1000 Q 252772 E  747232 H 105573 D  88290 I 53119 R  9121 C  9072 L  221K
      *** crawling 0.2 pps @ 0.0 Mbps
[243] 1000 Q 246862 E  753142 H 106441 D  89009 I 53371 R  9175 C  9127 L  222K
      *** crawling 0.2 pps @ 0.0 Mbps
[245] 1000 Q 240601 E  759403 H 107291 D  89711 I 53619 R  9223 C  9176 L  223K
      *** crawling 0.2 pps @ 0.0 Mbps
[247] 1000 Q 235341 E  764663 H 108049 D  90318 I 53839 R  9280 C  9227 L  223K
      *** crawling 0.2 pps @ 0.0 Mbps
[249] 1000 Q 229140 E  770864 H 109036 D  91154 I 54136 R  9325 C  9281 L  224K
      *** crawling 0.2 pps @ 0.0 Mbps
[251] 1000 Q 223381 E  776623 H 110018 D  91944 I 54386 R  9364 C  9309 L  224K
      *** crawling 0.1 pps @ 0.0 Mbps
[253] 1000 Q 218300 E  781704 H 110815 D  92586 I 54602 R  9391 C  9339 L  225K
      *** crawling 0.1 pps @ 0.0 Mbps
[255] 1000 Q 211794 E  788210 H 111769 D  93368 I 54880 R  9442 C  9390 L  226K
      *** crawling 0.2 pps @ 0.0 Mbps
[257] 1000 Q 203427 E  796577 H 112843 D  94256 I 55177 R  9503 C  9456 L  229K
      *** crawling 0.3 pps @ 0.0 Mbps
[259] 1000 Q 196330 E  803674 H 113820 D  95085 I 55489 R  9552 C  9500 L  229K
      *** crawling 0.2 pps @ 0.0 Mbps
[261] 1000 Q 190120 E  809884 H 114779 D  95882 I 55783 R  9600 C  9546 L  230K
      *** crawling 0.2 pps @ 0.0 Mbps
[263] 1000 Q 184830 E  815174 H 115610 D  96544 I 56019 R  9643 C  9595 L  231K
      *** crawling 0.2 pps @ 0.0 Mbps
[265] 1000 Q 178631 E  821373 H 116504 D  97254 I 56247 R  9691 C  9637 L  232K
      *** crawling 0.2 pps @ 0.0 Mbps
[267] 1000 Q 172602 E  827402 H 117394 D  97979 I 56518 R  9741 C  9688 L  233K
      *** crawling 0.2 pps @ 0.0 Mbps
[269] 1000 Q 165990 E  834014 H 118350 D  98767 I 56804 R  9784 C  9735 L  234K
      *** crawling 0.2 pps @ 0.0 Mbps
[271] 1000 Q 157754 E  842250 H 119468 D  99661 I 57119 R  9823 C  9773 L  234K
      *** crawling 0.1 pps @ 0.0 Mbps
```

```
[273] 1000 Q 149867 E  850137 H 120511 D 100493 I 57418 R  9887 C  9833 L  236K
      *** crawling 0.2 pps @ 0.0 Mbps
[275] 1000 Q 144641 E  855363 H 121220 D 101029 I 57622 R  9929 C  9879 L  238K
      *** crawling 0.2 pps @ 0.0 Mbps
[277] 1000 Q 138065 E  861939 H 122058 D 101671 I 57852 R  9962 C  9913 L  239K
      *** crawling 0.1 pps @ 0.0 Mbps
[279] 1000 Q 131248 E  868756 H 122933 D 102345 I 58112 R 10008 C  9954 L  240K
      *** crawling 0.1 pps @ 0.0 Mbps
[281] 1000 Q 123129 E  876875 H 123879 D 103083 I 58380 R 10059 C 10004 L  241K
      *** crawling 0.2 pps @ 0.0 Mbps
[283] 1000 Q 117031 E  882973 H 124771 D 103762 I 58659 R 10103 C 10046 L  241K
      *** crawling 0.1 pps @ 0.0 Mbps
[285] 1000 Q 111533 E  888471 H 125548 D 104370 I 58883 R 10140 C 10089 L  242K
      *** crawling 0.2 pps @ 0.0 Mbps
[287] 1000 Q 105980 E  894024 H 126324 D 104996 I 59126 R 10191 C 10136 L  242K
      *** crawling 0.2 pps @ 0.0 Mbps
[289] 1000 Q  99965 E  900039 H 127266 D 105740 I 59405 R 10239 C 10192 L  244K
      *** crawling 0.2 pps @ 0.0 Mbps
[291] 1000 Q  94506 E  905498 H 128077 D 106401 I 59661 R 10279 C 10229 L  244K
      *** crawling 0.1 pps @ 0.1 Mbps
[293] 1000 Q  87371 E  912633 H 129125 D 107235 I 59946 R 10328 C 10277 L  245K
      *** crawling 0.2 pps @ 0.0 Mbps
[295] 1000 Q  81001 E  919003 H 130067 D 107974 I 60201 R 10366 C 10317 L  246K
      *** crawling 0.1 pps @ 0.0 Mbps
[297] 1000 Q  75901 E  924103 H 130908 D 108671 I 60448 R 10402 C 10348 L  246K
      *** crawling 0.1 pps @ 0.0 Mbps
[299] 1000 Q  70881 E  929123 H 131700 D 109309 I 60680 R 10444 C 10390 L  249K
      *** crawling 0.1 pps @ 0.0 Mbps
[301] 1000 Q  64204 E  935800 H 132668 D 110097 I 60947 R 10491 C 10434 L  250K
      *** crawling 0.1 pps @ 0.0 Mbps
[303] 1000 Q  57195 E  942809 H 133569 D 110830 I 61208 R 10537 C 10484 L  251K
      *** crawling 0.2 pps @ 0.0 Mbps
[305] 1000 Q  49799 E  950205 H 134499 D 111560 I 61482 R 10591 C 10538 L  253K
      *** crawling 0.2 pps @ 0.0 Mbps
[307] 1000 Q  42455 E  957549 H 135498 D 112348 I 61764 R 10639 C 10584 L  254K
      *** crawling 0.1 pps @ 0.0 Mbps
[309] 1000 Q  35480 E  964524 H 136406 D 113075 I 62043 R 10689 C 10632 L  255K
      *** crawling 0.2 pps @ 0.0 Mbps
[311] 1000 Q  28099 E  971905 H 137284 D 113773 I 62298 R 10728 C 10678 L  256K
      *** crawling 0.1 pps @ 0.0 Mbps
[313] 1000 Q  21236 E  978768 H 138104 D 114425 I 62529 R 10781 C 10728 L  257K
      *** crawling 0.2 pps @ 0.0 Mbps
[315] 1000 Q  14458 E  985546 H 138897 D 115059 I 62759 R 10819 C 10769 L  258K
      *** crawling 0.1 pps @ 0.0 Mbps
[317]  806 Q      0 E 1000004 H 139300 D 115404 I 62892 R 10850 C 10803 L  258K
      *** crawling 0.1 pps @ 0.0 Mbps
[319]  503 Q      0 E 1000004 H 139300 D 115406 I 62892 R 10850 C 10805 L  258K
      *** crawling 0.0 pps @ 0.0 Mbps
[321]  374 Q      0 E 1000004 H 139300 D 115408 I 62894 R 10850 C 10805 L  258K
      *** crawling 0.0 pps @ 0.0 Mbps
[323]  304 Q      0 E 1000004 H 139300 D 115417 I 62901 R 10851 C 10806 L  258K
      *** crawling 0.0 pps @ 0.0 Mbps
[325]  227 Q      0 E 1000004 H 139300 D 115425 I 62907 R 10854 C 10809 L  258K
      *** crawling 0.0 pps @ 0.0 Mbps
[327]  164 Q      0 E 1000004 H 139300 D 115435 I 62910 R 10854 C 10809 L  258K
      *** crawling 0.0 pps @ 0.0 Mbps
[329]  121 Q      0 E 1000004 H 139300 D 115435 I 62910 R 10854 C 10809 L  258K
      *** crawling 0.0 pps @ 0.0 Mbps
[331]   92 Q      0 E 1000004 H 139300 D 115435 I 62910 R 10854 C 10809 L  258K
      *** crawling 0.0 pps @ 0.0 Mbps
[333]   58 Q      0 E 1000004 H 139300 D 115435 I 62910 R 10854 C 10809 L  258K
      *** crawling 0.0 pps @ 0.0 Mbps
[335]   30 Q      0 E 1000004 H 139300 D 115435 I 62910 R 10854 C 10809 L  258K
      *** crawling 0.0 pps @ 0.0 Mbps
[337]    7 Q      0 E 1000004 H 139300 D 115435 I 62910 R 10854 C 10809 L  258K
      *** crawling 0.0 pps @ 0.0 Mbps
[339]    6 Q      0 E 1000004 H 139300 D 115435 I 62910 R 10854 C 10809 L  258K
      *** crawling 0.0 pps @ 0.0 Mbps
[341]    5 Q      0 E 1000004 H 139300 D 115435 I 62910 R 10854 C 10809 L  258K
      *** crawling 0.0 pps @ 0.0 Mbps
[343]    5 Q      0 E 1000004 H 139300 D 115435 I 62910 R 10854 C 10809 L  258K
```

```
      *** crawling 0.0 pps @ 0.0 Mbps
[345]    2 Q      0 E 1000004 H 139300 D 115435 I 62910 R 10854 C 10809 L  258K
      *** crawling 0.0 pps @ 0.0 Mbps
[347]    2 Q      0 E 1000004 H 139300 D 115435 I 62910 R 10854 C 10809 L  258K
      *** crawling 0.0 pps @ 0.0 Mbps
[349]    0 Q      0 E 1000004 H 139300 D 115435 I 62910 R 10854 C 10809 L  258K
      *** crawling 0.0 pps @ 0.0 Mbps

Extracted 1000004 URLs @ 2865/s
Looked up 139300 DNS names @ 399 / s
Attempted 62910 site robots @ 180 / s
Crawled 10809 pages @ 30 / s (171.19 MB)
Parsed 258970 links @ 742 / s
HTTP codes : 2xx = 5464, 3xx = 1384, 4xx = 3839, 5xx = 122, other = 0
```

Google Graph Size Analysis

We will use the numbers from the Full trace (above), for all of the calculations we do in the report. In the trace we can see that there were, $n = 5464$, response codes in the 200 range, for which we parsed a total of, $L_{tot} = 258970$, links. Therefore we can find the average to be:

$$\overline{L} = \frac{L_{tot}}{n} = \frac{258970}{5464} = 47.40$$

Number of Edges

For the number of edges, we know:

- $N_{pages} = 10^{12}$, Google crawls 1T (trillion) pages

Therefore, we can find edges of the webgraph, $E = N_{pages} \times \overline{L}$:

$$E = 1,000,000,000,000 \times 47.40 = 47,400,000,000,000$$

So, the estimated number of edges in Google's webgraph is about 47.4 trillion.

Storage Size

For each page (node), we need to store:

- The page's own 64-bit hash (8 bytes)
- The 64-bit hashes (8 bytes) of all its out-neighbors

Now we can find the Storage per node, $S_{node} = 8 + 8 \times \overline{L}$:

$$S_{node} = 8 + (8 \times 47.40) = 387.2 \text{ bytes}$$

And total storage, $S_{total} = N_{pages} \times S_{node}$:

$$S_{total} = 1,000,000,000,000 \times 387.2 \text{ bytes}$$

$$= 387,200,000,000,000 \text{ bytes}$$

$$\approx 352.18 \text{ TB}$$

Therefore, we can estimate that Google's webgraph, assuming they crawl 1 trillion pages, would contain approximately 47.4 trillion edges and occupy about 352.18 terabytes of storage space.

Google Bandwidth

We know that the total bytes that we downloaded for this trace of the 1M output was 171.19MB. We can divide this by the total number of responses we got 10809 to find that the average size per response is:

$$\overline{Size} = \frac{171.19\text{ MB}}{10809\text{ pages}} \approx 0.01584\text{ MB/page}$$

With this, we can estimate the bandwidth that Google needs to be able to crawl 10 Billion pages daily:

- Pages to crawl per day: $N_{pages} = 10,000,000,000$.
- Seconds in a day: $s = 24 \times 60 \times 60 = 86,400$

$$\text{Pages per second} = \frac{N_{pages}}{s} = \frac{10,000,000,000}{86,400} \approx 115,741\text{ page/s}$$

Bandwidth required:

$$\text{Pages per second} \times \overline{Size} = 115,741\text{ page/s} \times 0.01584\text{ MB/page}$$

$$\approx 1,833\text{ MB/s} = 14,664\text{ Mbps} = 14.66\text{ Gbps}$$

Therefore, to crawl 10B pages a day, Google would need a bandwidth of approximately 14.66 Gigabits per second (Gbps).

Probability Analysis

- What is the probability that a link in the input file contains a unique host?

$$\frac{H}{E} = \frac{139300}{1000004} = 13.93\%$$

- What is the probability that a unique host has a valid DNS record?

$$\frac{D}{H} = \frac{115435}{139300} = 82.87\%$$

- What percentage of contacted sites had a 4xx robots file?

$$\frac{R}{I} = \frac{10854}{62910} = 17.25\%$$

Texas A&M In-Degree (`tamu.edu`)

For the in-degree of tamu.edu, I found that there were a total of 4 pages that linked to our domain: 3 from outside, and 1 from inside `tamu.edu`. To find this information, I iterated through the links returned by the `HTMLParserBase::Parse` function, and passed them to my Url parser. I modified my Url parser to check if the parsed URL's host ended with `tamu.edu`, if it did it would set a boolean flag.

This allowed me to check the parsed links on each page for connections to tamu.edu and then reference whether than link was an internal or external link to `tamu.edu`.

Extra Credit

My code fully supports HTTP 1.1 reponses, but I have it disabled by default to decrease processing time for benchmarking for the grade. This functionality can be enabled by simply opening the project, and uncommenting a single `#define` statement.

In the Precompiled Header file, line 17 (`pch.h:17`), there is the following:

```
#ifndef PCH_H
#define PCH_H

// Uncomment this to test HTTP1.1 chunking functionality
//#define HTTP1_1

//...
```

Simply uncomment it, and the HTTP 1.1 will be enabled:

```
#ifndef PCH_H
#define PCH_H

// Uncomment this to test HTTP1.1 chunking functionality
#define HTTP1_1

//...
```