Lab 6 Grading Rubric by Jonathan Yu and Mark Aligbe

All grades are calculated starting from 120 and points are deducted for each code that is not met. As with all labs, “make” *failing to produce an executable* will result in an automatic 0 for that part (60 point deduction). In the Makefile, we will ensure that your include and linking directories are correct, but points will be deducted if they are incorrect.

Table A. Rule Codes and Points

Code	Points	Description
1	**-10**	**Makefile Requirements**
1.1	-5	Incorrect dependencies / Errors in compilation or linking for Part 1
1.2	-5	Incorrect dependencies / Errors in compilation or linking for Part 2
2	**-10**	**Organization**
2.11	-3	Misnamed executable and/or misplaced files for Part 1
2.12	-3	Misnamed executable and/or misplaced files for Part 2
2.21	-2	README content missing (e.g. how to use your program if it takes non-standard input) for Part 1
2.22	-2	README content missing (e.g. how to use your program if it takes non-standard input) for Part 2
3	**-6**	**Commit Requirements (at least 5 meaningful commits for any credit)**
3.1	-3	Committed a functional Part 1
3.2	-3	Committed a functional Part 2
4	**-17**	**Program Requirements for Part 1**
4.1	-3	Fails to handle more than one client sequentially.
4.2	-4	Does not include a blank line after all the results or includes ‘lookup:’ when prompting the client.
4.3	-4	Does not return results when a query is submitted (e.g. not reading the client’s first query).
4.4	-6	The output does not match the solution.
5	**-23**	**Program Requirements for Part 2**

5.1	-5	Fails to connect to the remote server.
5.2	-8	Fails to parse the headers and take the appropriate action (e.g. displaying the 301 error in T3) or includes them in the output file.
5.3	-10	The output file is not named correctly or does not match the solution (excluding containing the HTTP headers).
6	**-54**	**Memory Errors**
6.11	-15	Improper uses of malloc / memory leaks (any bytes left on heap) in Part 1
6.12	-9	Improper uses of malloc / memory leaks (other than the still reachable bytes due to 'gethostbyname()' or 'getaddrinfo()' or the possible lost bytes due to 'getaddrinfo()') in Part 2
6.21	-15	Memory errors / Segmentation faults occur in Part 1
6.22	-15	Memory errors / Segmentation faults (other than the errors due to 'getaddrinfo()') occur in Part 2

Table B. Test Cases:

For http-client, testing was ignorant of trailing whitespace at the end of a file. That is to say, if a file contains 5 blank lines after the last printable character and your file contains no blank lines after the last printable character, they are considered equivalent. The base files were produced using Jae's http-client.c solution.
For mdb-lookup-server, to ensure consistency, a copy of the database file was made so that all results will be uniform across everyone tested. White space characters before and after a line are ignored.

http-client	**Arguments**	**Result**
T1	www2.warnerbros.com 80 /spacejam/movie/jam.htm	jam.htm
T2	www.gnu.org 80 /software/make/manual/make.html	make.html (file too big for pastebin, view the source code using your browser's functionality)
T3	www.cplusplus.com 80 /index.html	"HTTP/1.1 301 Moved Permanently"
mdb-lookup -server	**Test Case (tested against**	**Result**

	/home/jae/cs3157-pub/bin/mdb-cs3157 as of 4/15/2013 3:05 PM EST)	
T4	michelle	L1: 82: {michelle} said {i need to do homework} L2: 85: {michelle} said {derp}
T5	five	(blank)
T6	(return)	(480 results)