CS 345: Homework 1 (Due: Sep 23, 2014)

Rules. Individual homeworks; see Handout 1 (aka Syllabus). Hardcopies may be submitted no later than start of class the day they are due; electronic copies by NOON-time the same day.

Problem 1. (15 points) **Create your Web-page and establish Freshness of it**

1.A. Objectives. The objective of this problem is for you to create a web-page if you don't already have one, establish a link from your web-page to another page (i.e. an html file) that you most likely do not currently have, named `cs345f14.html`, and then from within that file create at a minimum a link to the course web-page.

(1.A.1) If you do not have a web-page, you will have the opportunity to create a web-page for yourself. This file is usually referred to as `index.html`. Whether you have such a page or not, we ask you that you minimally change it including the addtion of an anchor (link) to another file of yours `cs345f14.html`.

(1.A.2) File `cs345f14.html` will minimally contain a link to the course web-page `http://www.cs.njit.edu/~alexg/courses/cs345/index.html`. Your are free to add another information though it is not required.

(1.A.3) After you are done with (1.A.1) and (1.A.2) you submit the link information of your web-page to us through an online form recording the transaction number and time. The latter information will become part of your homework submission.

1.B UCID login and password. Usually the password you will need for AFS action below is the UCID login and password. If it does not work go to `http://ist.njit.edu/support/afs/afspassword.php` and use option 3 to make all of your passwords the same as a NEW UCID password. If you have forgotten your UCID password go to `http://ist.njit.edu/support/ucid` and one of the links in item 4 gives you the option of an Unattended Password Reset. Another option Global Password Change is similar to the operation of the AFS link above.

1.C Instructions to create a Web-page. Read the information at `http://ist.njit.edu/webhosting` if you have no web-page and do not otherwise know how to create one. You may need to login to an AFS machine through SSH (Secure Shell) such as `afsconnect1.njit.edu` or `afsconnect2.njit.edu`. You may also utilize file `template.html` that is provided in section B of the course web-page.

1.D Your Web-page (aka index.html). At `http://web.njit.edu/~UCID/index.html` after you complete the creation of your web-page there would be a file name`index.html` that you have just created. It must minimally include and conform to the following requirements.

1.D.1 File `index.html` has no explicit references to `CS 345` or to the intructor. It minimally lists your name, NJIT email address or UCID.

1.D.2 File `index.html` has a time stamp at the bottom recording some date and time information. This might be of the form `This file was last updated on`. See also template.html

1.D.3 File `index.html` has a properly formulated `TITLE` tag that includes text along the lines `Web-page of` including your name.

1.D.4 File `index.html` has a properly formulated `META` tag keyword list appropriate to your web-page content and your design. See template.html for an example.

1.D.5 File `index.html` has an anchor link with some relevant anchor text to a `cs345f14.html` file that would co-exist in the same directory with this `index.html` file.

1.E File `cs345f14.html`**.**
In the same location/directory as `http://web.njit.edu/~UCID/index.html` create a second html file using or not `template.html` of Section B of the course web-page as guidance. Its name should be `cs345f14.html` and minimally include and conform to the following requirements.

1.E.1 File `cs345f14.html` has an anchor that points to the course web-page.

1.E.2 File `cs345f14.html` has a time stamp at the bottom recording some date and time information. This might be of the form `This file was last updated on`. See also template.html.

1.E.3 File `cs345f14.html` has a properly formulated `TITLE` tag that includes text along the lines `Course Taking in Fall 2014` .

1.E.4 File `cs345f14.html` has a properly formulated `META` tag keyword list appropriate to your web-page content and your design and the link it provides. See template.html for an example.

1.E.5 File `cs345f14.html` has an anchor link with some relevant anchor text to the course web-page.

1.F Future Actions.

Check the Web (Google, Bing, etc) to find out whether you have been indexed. If you have try to retrieve a cached copy of it (click on the link and a Javascript pop up menu will appear).

Update periodically your web-page's `This file was last updated on` time stamp, so that you can find out when it was cached from the time-stamp information of yours, if Google or Bing do not provide their own information.

1.G Grading and Deliverables.

(a) You have created your web-page i.e. file `index.html` and then modified it accordingly per section 1.D to include certain information including an anchor/link to `cs345f14.html` .

(b) You have created `cs345f14.html` and modified it according to section 1.E.

(c) You have visited Section B of the course Web-page, and submitted link information to your web-page, recorded the submission's time and transaction number that you provided.

(d) Submit the transaction number and submission time, with the rest of the homework.

(e) Between now and later homeworks, track your page on the Web and record (screenshots) how often it gets updated using the cached (by a search engine) copy and the timestamp information recorded there. This would help you in a later assignment to collect more points by showing how often a page gets indexed by a search engine (aka freshness).

Problem 2. (15 POINTS) **Crawling and Wget**

2.A Wget.

The UNIX command wget available on an AFS machine allows you to fetch copies of web-pages. This includes a copy of the course web-page. However it is idiosyncratic and thus the prefix you give to it such as `cs.njit.edu` or `web.njit.edu` or `www.cs.njit.edu` will determine what you get or not. Login on afs (see also 1.C) and read information about wget by doing `man wget` under AFS or `wget --help`. On UNIX typing `du -s dirname` you can get info about the file size of all files in the directory (and subdirectories) named `dirname`. The size may be in kilobytes or mutliples of 512B (read the manual). Certain details are missing because you are expected to read the manual pages and also to improvise. Wget generates a log file while transferring file. You can capture it into a file by redirecting its output to a file named `capturefilename` as partially shown in the following invocation fragment

```
wget ....   alexg/courses/cs345/index.html > & capturefilename
```

Moreover, it creates a directory that stores all the relevant files.

2.B Wget and a copy of the course Web-page.

Grab a copy of the course web-page. Among the three various prefixes of a URL listed above one would get what you wish for. The total size of files that you would transfer should be at least 3.6MiB rather than closer to 400MiB. (MiB is SI notation for Mebi-Bytes.) Use appropriate options to get the desired result!

Then answer the following questions

2.B.1 Give the size in MiB of the downloaded files (must be more than few Megabytes).

2.B.2 Give the number of downloaded documents (as reported in the log files). Note relevant requests by wget are reported accordingly so this information is available in the log file captured. (the number you report must be much greater than thirty.)

2.B.3 How many URLs were not found, i.e. even if there was a link to them, the corresponding file was not found. (In the capture file, this generates a relevant error.)

2.B.4 How many URLs resulted in rejects because of password authentication issues? (In the capture file, this generates a relevant error.)

2.B.5 What kind of error was generated for [2.B.3] and what for [2.B.4]?

Problem 3. (15 POINTS) **Webcrawler**

3.A Download, Unzip, Run.

Download the zip file **webcrawler_2_1.zip**. It is available in the handouts section of the course web-page. Unzip it by typing `unzip webcrawler_2_1.zip` and it will unfold 7 Python files. You will run Python on the same machine you used for the previous problem. There are several versions of Python on AFS but the one that works correctly is version 2.6. Thus your invocation will involve something along the lines

`/usr/bin/python2.6 modular_crawler.py`

There are some additional options for the crawler: we call them second, third etc, where option one is just the name `modular_crawler.py` and the python interpreter does not count at all. The second option is the domain restriction, the third option the starting page and the last option a 0 to mean no restriction or a page bound on the number of pages to be traversed. Output can be captured into a file.

3.B Test 1: Warmup

```
/usr/bin/python2.6 modular_crawler.py web.njit.edu http://web.njit.edu/~alexg/courses/cs345/index.html 10000
```

Run the following version. How many links are found that are in the domain as specified in the second? Such links are clearly reported in the output that can be captured and a summary is given at the end of that output/report as well.

3.C Test 2: cs.njit.edu

Is there a difference if cs.njit.edu is used instead of web.njit.edu in the command line? Report differences and number of pages found.

3.D Test 3: 3.B and 3.C repeated

Repeat the two tests before but for the domain `www.njit.edu` (second option for both cases), even if the third option remains as is in Test 1 (3.B) and Test 2 (3.C) respectively.

Did you have a 10000 cutoff in any of the two cases? Report number of pages found.

3.E Is there a URL that contains the number sequence

Is there a URL that contains the number sequence `47182`? What does it relate to?

Problem 4. (15 POINTS) **(TO BE OR NOT TO BE: Google vs Bing face off)**
The brackets below ⟨ and ⟩ indicate the start and end of a query. They are not typed in the query. The quotation marks are typed (eg in query 9). We queried the two search engines on Fri Sep 5, 2014 and we got the following results. Number of hits reported are in millions.

If the query systems of the two search engines are accepting Boolean queries,

(i) Within the GOOGLE results do you observe any discrepancy? More than one? Explain it/them.

(ii) Within the BING results do you observe any discrepancy? More than one? Explain it/them.

(iii) Based on total number of discrepancis involved, which of the two is more reliable for Boolean-type queries?

No	Query Terms	GOOGLE	BING
01	⟨to⟩	5760	13600
02	⟨be⟩	1770	2780
03	⟨or⟩	1690	3810
04	⟨not⟩	1580	9210
05	⟨to be⟩	24900	1620
06	⟨to be or not⟩	13200	9220
07	⟨or not to be⟩	13200	9220
08	⟨not to be⟩	16800	9220
09	⟨"to be or not to be"⟩	1.16	47
10	⟨to be OR not to be⟩	25270	55
11	⟨to be or not to be⟩	279	9220
11	⟨to be to be or not⟩	13200	9220
11	⟨to be or not or not⟩	13200	9220

Table 1: Table for Problem 4

Date Posted: 9/05/2014