Assignment 1

Fall 2014
CS595 Web Science
Dr. Michael Nelson

Mathew Chaney

September 2, 2014

Contents

List of Figures

1 Question 1

1.1 Question

Demonstrate that you know how to use "curl" well enough to correctly POST data to a form. Show that the HTML response that is returned is "correct". That is, the server should take the arguments you POSTed and build a response accordingly. Save the HTML response to a file and then view that file in a browser and take a screen shot.

1.2 Resources

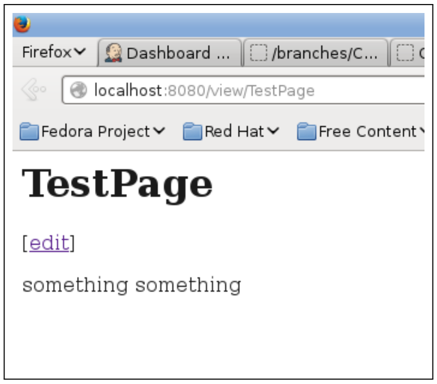
- LaTeX: http://www.electronics.oulu.fi/latex/examples/example_1
- LaTeX: http://scott.sherrillmix.com/blog/programmer/displaying-code-in-latex/
- curl: http://curl.haxx.se/docs/httpscripting.html#POST

1.3 Answer

Using a simple wiki server built from the Go language net/http package tutorial, found here: https://golang.org/doc/articles/wiki/

```
[mchaney@mchaney-d gowiki]$ curl -v -d "body=something something" localhost:8080/save/TestPage
* About to connect() to localhost port 8080 (#0)
*   Trying ::1...
* connected
* Connected to localhost (::1) port 8080 (#0)
> POST /save/TestPage HTTP/1.1
> User-Agent: curl/7.27.0
> Host: localhost:8080
> Accept: */*
> Content-Length: 24
> Content-Type: application/x-www-form-urlencoded
>
* upload completely sent off: 24 out of 24 bytes
< HTTP/1.1 302 Found
< Location: /view/TestPage
< Date: Fri, 29 Aug 2014 13:32:01 GMT
< Content-Length: 0
< Content-Type: text/plain; charset=utf-8
<
* Connection #0 to host localhost left intact
* Closing connection #0
```

The wiki uses the `/save/` url to save the contents of a text box as the body of the target page.



TestPage

[edit]

something something

Figure 1: Screenshot of curl POST result

2 Question 2

2.1 Question

Write a Python program that:

1. takes one argument, like "Old Dominion" or "Virginia Tech"
2. takes another argument specified in seconds (e.g., "60" for one minute).
3. takes a URI as a third argument:
   http://sports.yahoo.com/college-football/scoreboard/
   or
   http://sports.yahoo.com/college-football/scoreboard/?week=2&conf=all
   or
   http://sports.yahoo.com/college-football/scoreboard/?week=1&conf=72
   etc.
4. dereferences the URI, finds the game corresponding to the team argument, prints out the current score (e.g., "Old Dominion 27, East Carolina 17), sleeps for the specified seconds, and then repeats (until control-C is hit).

2.2 Resources

I used BeautifulSoup and Requests to write a bot that plays the game http://www.kingdomofloathing.com/. Code for which can be found @ http://www.github.com/mattchaney/meatmachine/. Both of these modules were useful for this assignment.

- Requests: http://docs.python-requests.org/en/latest/
- BeautifulSoup: http://www.crummy.com/software/BeautifulSoup/bs4/doc/

2.3 Source

```python
#!/usr/bin/env python

import sys
import requests
import time
import re
from bs4 import BeautifulSoup

if __name__ == "__main__":
    if len(sys.argv) != 4:
        print "Usage:\n\tpython getscore.py [school] [seconds] [uri]\n"
        sys.exit()
    school = sys.argv[1]
    period = float(sys.argv[2])
    uri = sys.argv[3]
    while True:
        soup = BeautifulSoup(requests.get(uri).content)
        gametag = soup.find('em', text=re.compile('^'+school+'$'))
        if not gametag:
            print "Game not found"
            sys.exit()
        game = gametag.parent.parent.parent
        teams = [game.find_all('span', {'class':'team'})[0].em.text, game.find_all('span', {'class':'team'})[1].em.text]
        scores = [game.find('span', {'class':'away'}).text, game.find('span', {'class':'home'}).text]
        print teams[0] + ' ' + scores[0] + ', ' + teams[1] + ' ' + scores[1]
        time.sleep(period)
```

2.4 Answer

```
[mchaney@mchaney-d q2]$ python getscore.py "Texas A&M" 10 "http://sports.yahoo.com/college-football/scoreboard/"
Texas A&M 52, South Carolina 28
Texas A&M 52, South Carolina 28
^CTraceback (most recent call last):
  File "getscore.py", line 26, in <module>
    time.sleep(period)
KeyboardInterrupt
[mchaney@mchaney-d q2]$ python getscore.py "alskds" 5 "http://sports.yahoo.com/college-football/scoreboard/"
Game not found
[mchaney@mchaney-d q2]$ python getscore.py "Texas A&M" "http://sports.yahoo.com/college-football/scoreboard/"
Usage:
    python getscore.py [school] [period] [uri]

[mchaney@mchaney-d q2]$
```

3 Question 3

3.1 Question

Consider the "bow-tie" graph in the Broder et al. paper (fig 9): http://www9.org/w9cdrom/160/160.html

Now consider the following graph:

```
A --> B
B --> C
C --> D
C --> A
C --> G
E --> F
G --> C
G --> H
I --> H
I --> J
I --> K
J --> D
L --> D
M --> A
M --> N
N --> D
```

3.2 Resources

- Graph Structure in the web: http://www9.org/w9cdrom/160/160.html
- Stanford, The web graph: http://nlp.stanford.edu/IR-book/html/htmledition/the-web-graph-1.html
- Notes from the class:
  SCC: Strongly Connected Component - all contained nodes are interconnected
  IN: Connects into SCC, but not out from SCC
  OUT: Connects out from SCC, but not in to SCC
  Tendril: In or out excluding all SCC
  Tube: IN->OUT or OUT->IN connection
  Disconnected: Not connected to other sites

3.3 Answer

For the above graph, give the values for:

IN: A, B, C, G
SCC: M
OUT: D, H
Tendrils: L, K, I, J
Tubes: N
Disconnected: E, F