CS 532: Assignment 1

Dinesh Kumar Paladhi

Spring 2016

Contents

1 Problem 1

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.

1.1 Solution

1. Started with understanding what curl does from the secure shell by using curl –help command.
2. Got various options which are used in curl. Later,Referred few sites in order to get in touch with curl.
3. My next step was to get the web forms which use POST method but It was hard to find them. Later, I found the link in the goggle groups and started using it.
4. I selected the following web page to post my data.

   ```
   http://www.cs.tut.fi/cgi-bin/run/~jkorpela/echo.cgi
   ```

5. The curl command i used to post data to a form is

   ```
   Curl --data "comments=Good to see you"
   http://www.cs.tut.fi/cgi-bin/run/~jkorpela/echo.cgi -o output.html
   ```

6. In the beginning i did not use -o output.html and so i got the html code as output on my command prompt.
7. curl -o output.html command creates a html file in the Z drive.

1.2 Results

1. Screen shot of the web page with the form to which I posted data is shown below.

Figure 1: Form in the Web page

2. The response I got after implementing the curl command is shown below

```
atria:~> curl --data "comments=Good to see you" http://www.cs.tut.fi/cgi-bin/run
/~jkorpela/echo.cgi -o output.html
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   425    0   401  100    24    300     17  0:00:01  0:00:01 --:--:--   300
```

Figure 2: Response for the curl command

3. Screen shot of the html file in the browser is shown below.

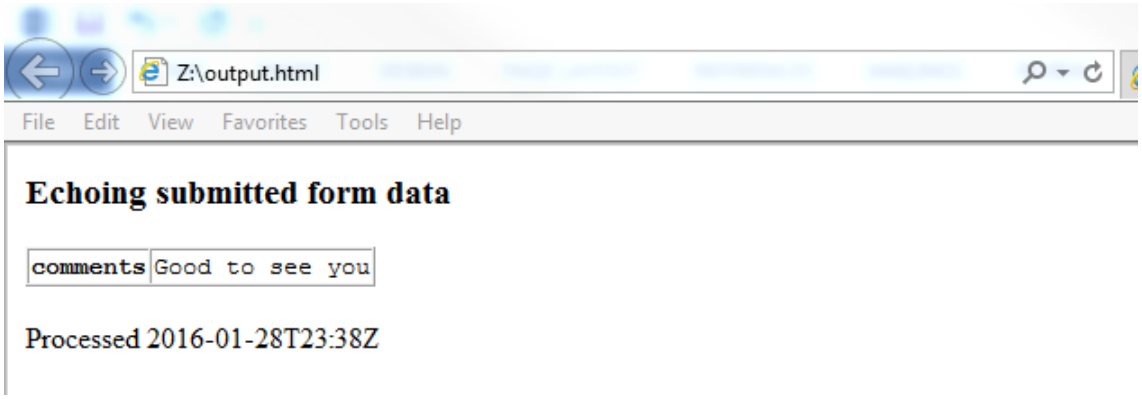

Figure 3: Response from the browser

2 Problem 2

Write a Python program that:

1. takes as a command line argument a web page
2. extracts all the links from the page
3. lists all the links that result in PDF files, and prints out the bytes for each of the links. (note: be sure to follow all the redirects until the link terminates with a "200 OK".)
4. show that the program works on 3 different URIs, one of which needs to be:
   `http://www.cs.odu.edu/~mln/teaching/cs532-s16/test/pdfs.html/`

2.1 Solution

1. First I tried to write the program and gather all the links from one particular link like www.cs.odu.edu.
2. For doing this I imported the required libraries.
3. Extracted the code using BeautifulSoup library.
4. Read the entire data from the above url using urllib2 and Beautifulsoup libraries.
5. Got all the data within the anchor tag using beautifulsoup function.
6. Wrote a sample function which checks for the url and returns false if it is not.
7. Now I ran a for loop where it stores the headers of each link and then also checks whether the content-type in the headers is pdf or not. If it is pdf then I am printing the url, content-length and the status code.
8. Similar process is carried on with 2 other links and the results are shown below.

2.2 Results

1. To get the sample output 1 the following command should be executed

   ```
   python prob2.py http://www.cs.odu.edu
   ```

```
atria:~/Webscience/cs532-s16/Assignment 1/Latex> python prob2.py http://www.cs.odu.edu
link    Content length     Response Code
http://www.cs.odu.edu/studentappointmentinfo.pdf     636560    200
http://www.cs.odu.edu/StrategicPlan0515_2010.pdf     909323    200
http://www.cs.odu.edu/files/cs_systems_services.pdf     412031    200
http://www.cs.odu.edu/files/csintroductioninfo.pdf     564602    200
atria:~/Webscience/cs532-s16/Assignment 1/Latex>
```

Figure 4: Output 1

2. To get the sample output 2 the following command should be executed

   `python prob2.py http://www.cs.odu.edu/~mln/teaching/cs532-s16/test/pdfs.html`

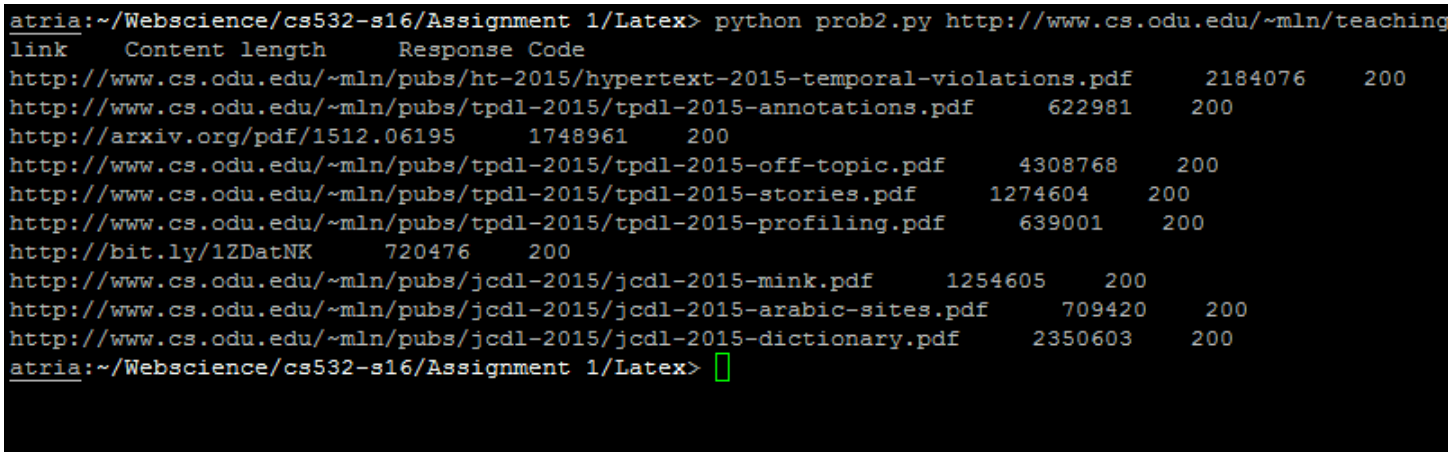

Figure 5: Output 2

3. To get the sample output 3 the following command should be executed

   `python prob2.py http://www.oracle.com/index.html`

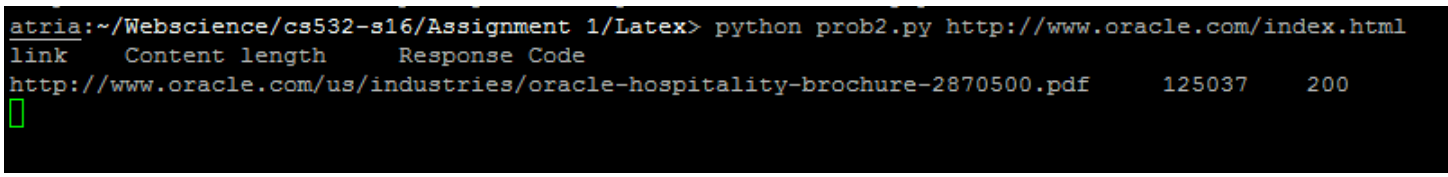

Figure 6: Output 3

2.3 Code Listing

```
import urllib2
import requests
import sys
from bs4 import BeautifulSoup


def valid_url(url):   #Function to eliminate href's which are not url's
    try:
        urllib2.urlopen(url)
        return True
    except Exception, e:
        return False

url = urllib2.urlopen(sys.argv[1]).read()  #Reads the entire data from the url
    given in the command prompt
soup = BeautifulSoup(url)
all_links= soup.find_all('a',href=True) # getting only anchor tag data
print "link","  ","Content length","    ","Response Code"  # Supportive text
for line in soup.find_all('a'):
                hrefLink = line.get('href')
                if valid_url(hrefLink) : #Calling the function defined above
                        response=urllib2.urlopen(hrefLink)
                        link_header=response.headers
                        type=link_header["Content-type"]
                        if type=="application/pdf":  #Checking whether the
                            file is pdf or not
                                print hrefLink,"    ", link_header["content-
                                    length"],"  ",response.getcode()
```

3 Problem 3

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
O --> A
P --> G
```

For the above graph, give the values for:
IN:
SCC:
OUT:
Tendrils:
Tubes:
Disconnected:

3.1 Solution

Model graph is drawn based on the bow-tie concept including all the node points stated above.

```
IN: M,O,P
SCC: A,B,C,G,
OUT: D,H
In Tendril: None
Out Tendril : L,K,I,J
Tubes: N
Disconnected: E,F
```

Explanations for each node have been provided below.

IN: O, M, P

According to the definition of IN, any node in IN can be connected to any other node in IN or can be connected to any node in SCC but not with any other nodes. M,O,P falls into this category they are connected to A,G nodes in SCC.

SCC: A, B, C, G

Strongly Connected Component consists of group of nodes which are connected to each other directly or indirectly and these nodes can be connected from IN and connected to OUT. A,B,C,G come under this category.

OUT: D, H

Nodes which exit from SCC and are not connected back to SCC come under OUT. D and H show these characteristics so they come under OUT.

Tendrils: I, J, K, L

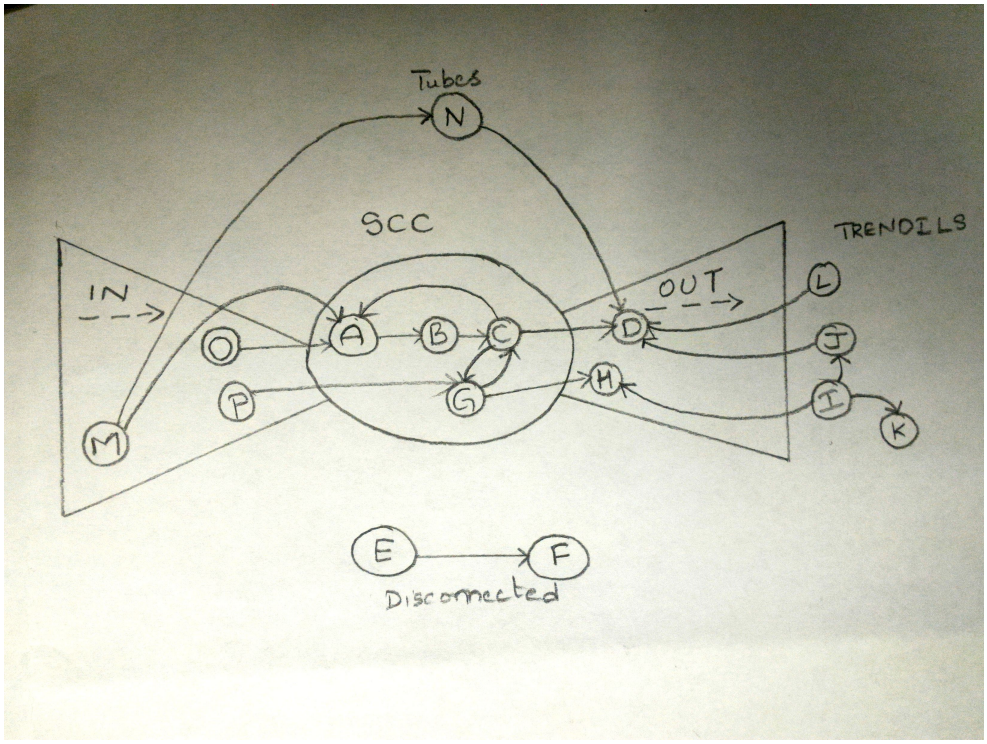

Figure 7: Drawing of the example graph

The nodes which cannot be directly connected to the nodes in SCC. I,J,K,L come under *Tendrils*.

Tubes: N

Nodes which pass from IN to OUT without any interaction with SCC are called *tubes*. N is a tube linking from M to D.

Disconnected: E, F

The Disconnected components link to no one in the graph, and stand alone. They are not defined explicitly by Broder, et. al, but their meaning is implied within the paper.

Bibliography

[1] How to fetch Internet Resources Using urllib2,
https://docs.python.org/2/howto/urllib2.html.

[2] Curl,
http://curl.haxx.se/docs/manpage.html.

[3] BeautifulSoup Documentation,
http://www.crummy.com/software/BeautifulSoup/bs4/doc/.

[4] Stack Overflow answering questions on Content-type,
http://stackoverflow.com/questions/2143674/how-do-i-get-a-content-type-of-a-file-in-python-with-url.

[5] Graph Strducture in the Web,
http://www9.org/w9cdrom/160/160.html .