Web Science: Assignment #1

Alexander Nwala

Puneeth Bikkasandra

Sunday, January 28, 2018

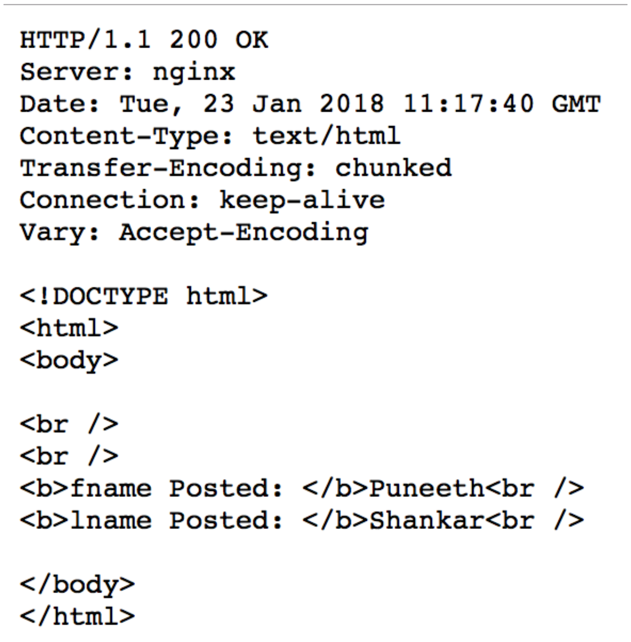
Contents

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.

SOLUTION :

```
curl -i -d "fname=Puneeth \& lname=Shankar" -X POST
http://www.cs.odu.edu/\˜anwala/files/temp/namesEcho.php
```



```
HTTP/1.1 200 OK
Server: nginx
Date: Tue, 23 Jan 2018 11:17:40 GMT
Content-Type: text/html
Transfer-Encoding: chunked
Connection: keep-alive
Vary: Accept-Encoding

<!DOCTYPE html>
<html>
<body>

<br />
<br />
<b>fname Posted: </b>Puneeth<br />
<b>lname Posted: </b>Shankar<br />

</body>
</html>
```

Figure 1: Sample ’curl’ with POST

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-s17/test/pdfs.html

SOLUTION

The solution for this problem is outlined by the following steps:

1. **Command Line Arguments :** Programmatically referred as **sys.argv** contains a list of arguments, which could be passed to the program. The below command takes 3 web pages as the arguments.

   ```
   python Assignment1.py http://www.google.com  https://www.facebook.com
    http://www.cs.odu.edu/˜mln/teaching/cs532-s17/test/pdfs.html
   ```

2. **Extracting the Links from the Web Pages :** The below code in Listing 1; extracts the web links, captures the redirection and finally saves in an independent file.

Listing 1: Assignment1.py

```
import requests
import sys
from BeautifulSoup import BeautifulSoup
links = []
def getURL(page):

    start_link = page.find("a href")
    if start_link == -1:
        return None, 0
    start_quote = page.find('"', start_link)
    end_quote = page.find('"', start_quote + 1)
    url = page[start_quote + 1: end_quote]
    return url, end_quote

def checkForRedirection(link1):
    response = requests.get(link1)
    if('Response [200]' in response):
        return link1
    else:
        return response.url

for webpage in sys.argv:
```

```
    baseUrl = webpage
    if(webpage == 'Assignment1.py'):
        continue
    else:
        response = requests.get(webpage)
        page = str(BeautifulSoup(response.content))
        while True:
            url, n = getURL(page)
            page = page[n:]
            if url:
                if('://' in url):
                    url = checkForRedirection(url)
                    links.append(url)
                else:
                    url = checkForRedirection(baseUrl + url)
                    links.append(url)
            else:
                break

pdf = open("pdfLinks.txt","w")
justLinks = open("justLinks.txt","w")
for link in links:
    response = requests.get(link)
    if(response.headers['content-type'] == 'application/pdf'):
        size = response.headers['content-Length']
        line = pdf.write(link+"      : Length - "+size+" bytes")
        line = pdf.write('\n')
        line = justLinks.write(link)
        line = justLinks.write('\n')
    else:
        line = justLinks.write(link)
        line = justLinks.write('\n')
pdf.close()
justLinks.close()
```

Links obtained after extraction:

Listing 2: Extracted Links

```
http://www.google.com/history/optout?hl=en&nzb=1
https://www.google.com/preferences?hl=en
https://www.google.com/advanced_search?hl=en&amp;authuser=0
https://translate.google.com/
https://www.google.com/intl/en/ads/
https://www.google.com/services/
https://plus.google.com/116899029375914044550
https://www.google.com/intl/en/about/
https://www.google.com/intl/en/policies/privacy/
https://www.google.com/intl/en/policies/terms/
https://www.google.com/history%5C
https://twitter.com/webscidl
http://www.dlib.org/dlib/november15/vandesompel/11vandesompel.html
https://arxiv.org/abs/1508.02315
https://arxiv.org/abs/1508.02315
http://www.cs.odu.edu/~mln/pubs/ht-2015/hypertext-2015-temporal-violations.pdf
```

Problem 2 continued on next page. . .

```
http://www.cs.odu.edu/~mln/pubs/tpdl-2015/tpdl-2015-annotations.pdf
https://arxiv.org/pdf/1512.06195.pdf
http://www.cs.odu.edu/~mln/pubs/tpdl-2015/tpdl-2015-off-topic.pdf
http://www.cs.odu.edu/~mln/pubs/tpdl-2015/tpdl-2015-stories.pdf
http://www.cs.odu.edu/~mln/pubs/tpdl-2015/tpdl-2015-profiling.pdf
https://link.springer.com/article/10.1007%2Fs00799-015-0150-6
http://www.cs.odu.edu/~mln/pubs/jcdl-2014/jcdl-2014-brunelle-damage.pdf
https://arxiv.org/abs/1506.06279
https://link.springer.com/article/10.1007%2Fs00799-015-0155-1
http://www.cs.odu.edu/~mln/pubs/jcdl-2015/jcdl-2015-temporal-intention.pdf
http://www.cs.odu.edu/~mln/pubs/jcdl-2015/jcdl-2015-mink.pdf
http://www.cs.odu.edu/~mln/pubs/jcdl-2015/jcdl-2015-arabic-sites.pdf
http://www.cs.odu.edu/~mln/pubs/jcdl-2015/jcdl-2015-dictionary.pdf
http://www.cs.odu.edu/~mln/teaching/cs532-s16/test/pdfs.html
https://link.springer.com/article/10.1007%2Fs00799-015-0140-8
https://www.facebook.com/
https://www.facebook.com/login/identify?ctx=recover&lwv=110
https://www.facebook.com/legal/terms
https://www.facebook.com/about/privacy
https://www.facebook.com/policies/cookies/
https://www.facebook.com/
https://www.facebook.com/
https://www.facebook.com/
https://www.facebook.com/pages/create/?ref_type=registration_form
https://www.facebook.com/r.php
https://www.facebook.com/login/
https://www.messenger.com/
https://www.facebook.com/lite/
https://www.facebook.com/mobile/?ref=pf
https://www.facebook.com/login.php?next=https%3A%2F%2Fwww.facebook.com%2Ffriends%2F
requests%2F%3Ffcref%3Dffi
https://www.facebook.com/directory/people/
https://www.facebook.com/directory/pages/
https://www.facebook.com/places/
https://www.facebook.com/games/
https://www.facebook.com/directory/places/
https://www.facebook.com/directory/celebrities/
https://www.facebook.com/directory/marketplace/
https://www.facebook.com/directory/groups/
https://www.facebook.com/recipes/
https://www.facebook.com/sport/
https://www.facebook.com/look/directory/
http://l.facebook.com/l.php?u=http%3A%2F%2Fmomentsapp.com%2F&h=
ATNlpLgX5yJaZnjeaPpE38-uHfNpVIy1Wn-IRryYPQto5Gw41XE0CzOpmUqk9fCffNvOZfmhe0BQOiC18Mt
qfiQ1o-LMv0K2Ug
https://l.facebook.com/l.php?u=https%3A%2F%2Finstagram.com%2F&h=
ATObPqFwFgsJIf_VnA6PyZ7CrCKE6bMtolqFkXQ9nXO5fxlWlT1dkU3pbIGDqrs_ccWvXKPTF8fQR4bd62g
QGTuo_ZVXmYEMqA
https://www.facebook.com/local/lists/245019872666104/
https://www.facebook.com/facebook
https://www.facebook.com/?placement=pflo
https://www.facebook.com/pages/create/?ref_type=sitefooter
https://developers.facebook.com/?ref=pf
```

```
https://www.facebook.com/careers/?ref=pf
https://www.facebook.com/policies/cookies/
https://www.facebook.com/help/?ref=pf
```

3. **Links with PDF Files :** The above code in Listing 1; segregates links to PDF files in to another file.

Listing 3: PDF files with their Length

```
http://www.cs.odu.edu/~mln/pubs/ht-2015/hypertext-2015-temporal-violations.pdf
Length - 2184076 bytes
http://www.cs.odu.edu/~mln/pubs/tpdl-2015/tpdl-2015-annotations.pdf
Length - 622981 bytes
https://arxiv.org/pdf/1512.06195.pdf
Length - 1748961 bytes
http://www.cs.odu.edu/~mln/pubs/tpdl-2015/tpdl-2015-off-topic.pdf
Length - 4308768 bytes
http://www.cs.odu.edu/~mln/pubs/tpdl-2015/tpdl-2015-stories.pdf
Length - 1274604 bytes
http://www.cs.odu.edu/~mln/pubs/tpdl-2015/tpdl-2015-profiling.pdf
Length - 639001 bytes
http://www.cs.odu.edu/~mln/pubs/jcdl-2014/jcdl-2014-brunelle-damage.pdf
Length - 2205546 bytes
http://www.cs.odu.edu/~mln/pubs/jcdl-2015/jcdl-2015-temporal-intention.pdf
720476 bytes
http://www.cs.odu.edu/~mln/pubs/jcdl-2015/jcdl-2015-mink.pdf
Length - 1254605 bytes
http://www.cs.odu.edu/~mln/pubs/jcdl-2015/jcdl-2015-arabic-sites.pdf
Length - 709420 bytes
http://www.cs.odu.edu/~mln/pubs/jcdl-2015/jcdl-2015-dictionary.pdf
Length - 2350603 bytes
```

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:

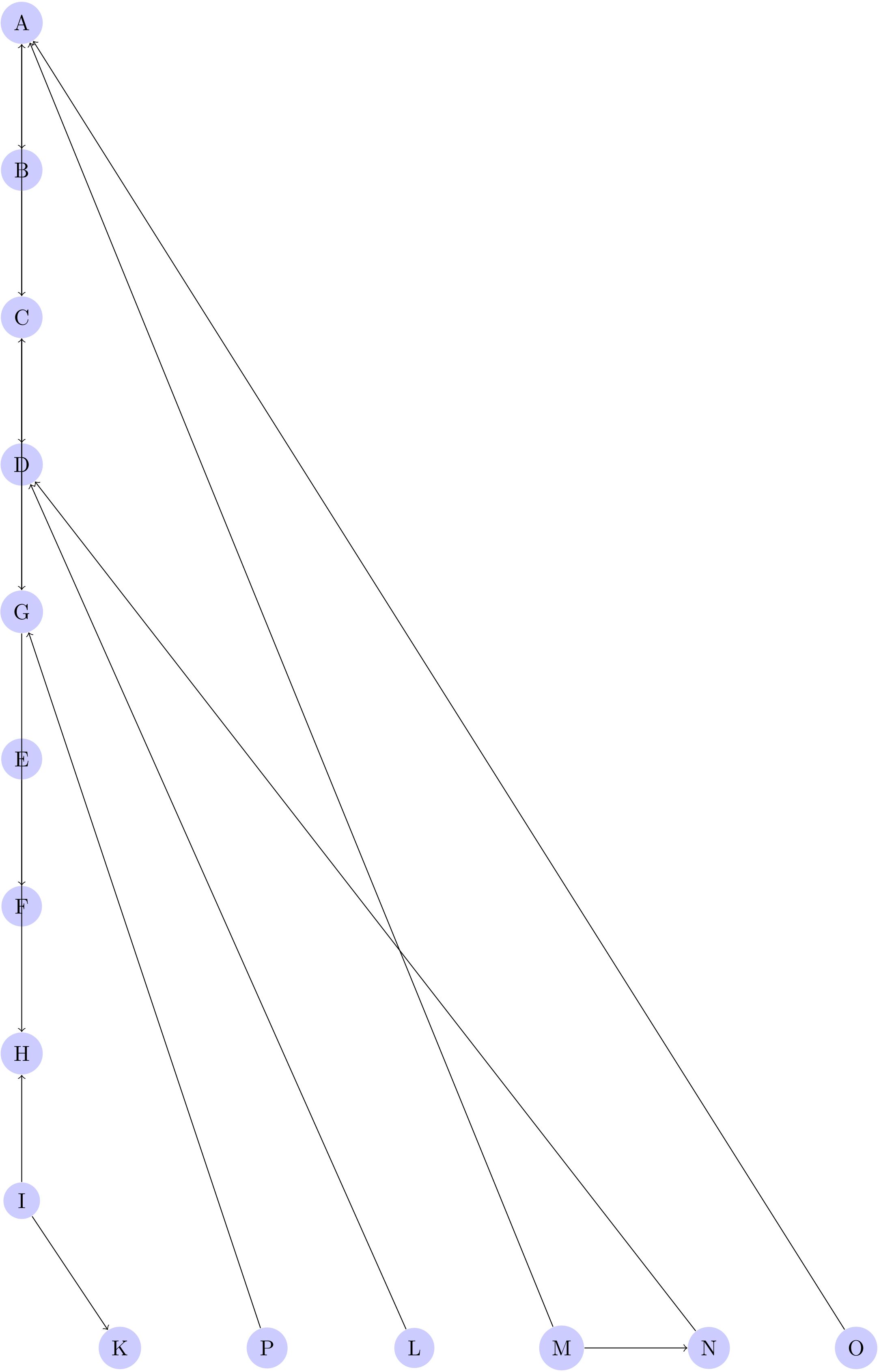
```
A --> B
B --> C
C --> D
C --> A
C --> G
E --> F
G --> C
G --> H
I --> H
I --> K
L --> D
M --> A
M --> N
N --> D
O --> A
P --> G
```

For the above graph, give the values for:

```
IN:
SCC:
OUT:
Tendrils:
Tubes:
Disconnected:
```

Problem 3 continued on next page. . .

SOLUTION



Problem 3 continued on next page...

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