

Assignment 1

CS 532: Introduction to Web Science

Spring 2017

Grant Atkins

Finished on January 25, 2017

1

Question

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.

Answer

The curl command is capable of solving this problem multiple ways. As stated by the Curl manual page, curl offers two options to post data:

- -F, -form, 'type='
- -d, -data, 'type='

The difference between the two is the content-type, where -F is multipart/form-data and -d is application/x-www-form-urlencoded [3]. Meaning -F can send files and parameters, while -d can just be used to send parameters via HTTP post.

For simplicity, I chose the latter route and used -d as part of my curl commands. I also chose to include -o, -output, which outputs the response to a file. The commands are as follows:

```
1 curl -d 'name=Grant Atkins' -d 'note=Praise Web Science' http://  
  www.cs.odu.edu/~gatkins/cs532/curlPost.php -o output/  
  correctResponse.html  
2  
3 curl http://www.cs.odu.edu/~gatkins/cs532/curlPost.php -o output  
  /incorrectResponse.html
```

Listing 1: Curl with and without post parameters

The first command sends two parameters in a post request to a URI, more specifically a PHP file that I created in my personal public html directory on the ODU Computer Science servers. The PHP file, as shown below in Listing 2, expects two parameters which are: name and note. Those two parameters

are then included inside of the html document response to show they were posted correctly, also the banner in which they are displayed should turn green if posted correctly like shown in Figure 1. The second command shows a curl command without post parameters to the same URI. This should show a red banner with an insult on your use of curl like shown in Figure 2.

```
1 <?php
2 $message = "";
3 $note = "";
4 $color = "purple";
5 if (isset($_POST["name"]) && isset($_POST["note"])) {
6     $message = "You rock at curl ".$_POST["name"];
7     $note = $_POST["note"];
8     $color = "green";
9 } else {
10     $message = "You suck at curl";
11 }
12
13 ?>
14 <html lang="en">
15 <head>
16     <title>CurlPost Example</title>
17     <meta charset="utf-8">
18     <meta name="viewport" content="width=device-width, initial-
19         scale=1">
20     <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/
21         bootstrap/3.3.7/css/bootstrap.min.css">
22     <script>
23     </script>
24 </head>
25 <body>
26     <div class="jumbotron text-center" style="background-color:<?
27         php echo $color; ?>;color:white;">
28         <h1><?php echo $message; ?></h1>
29         <h2><?php echo $note; ?></h2>
30     </div>
31 </body>
```

Listing 2: PHP Script for receiving form post parameters

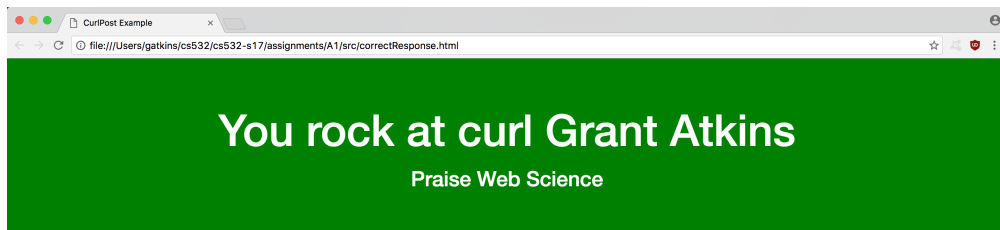


Figure 1: Correct response rendered in browser

```
1 <html lang="en">
2 <head>
3   <title>CurlPost Example</title>
4   <meta charset="utf-8">
5   <meta name="viewport" content="width=device-width, initial-
6     scale=1">
7   <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/
8     bootstrap/3.3.7/css/bootstrap.min.css">
9   <script>
10  </script>
11 </head>
12 <body>
13   <div class="jumbotron text-center" style="background-color:
14     green; color: white;">
15     <h1>You rock at curl Grant Atkins</h1>
16     <h2>Praise Web Science</h2>
17   </div>
18 </body>
```

Listing 3: Correct response html content outputted by curl command

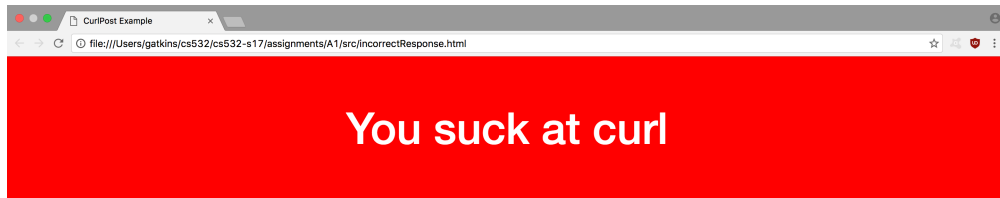


Figure 2: Incorrect response rendered in browser

```
1 <html lang="en">
2 <head>
3   <title>CurlPost Example</title>
4   <meta charset="utf-8">
5   <meta name="viewport" content="width=device-width, initial-
      scale=1">
6   <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/
      bootstrap/3.3.7/css/bootstrap.min.css">
7   <script>
8   </script>
9 </head>
10
11 <body>
12   <div class="jumbotron text-center" style="background-color:red
      ;color:white;">
13     <h1>You suck at curl</h1>
14     <h2></h2>
15   </div>
16 </body>
```

Listing 4: Incorrect response html content outputted by curl command

2

Question

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>

Answer

```
1  #!/usr/bin/env python
2
3  import sys
4  from bs4 import BeautifulSoup
5  from urllib2 import urlopen, HTTPError, URLError, Request
6  from urlparse import urljoin, urlparse
7  from httplib import BadStatusLine
8
9
10 def findPdfs(html, baseurl):
11     """
12     Take html string as parameter and parse through links ('a'
13     elements). Print final redirect url and bytes
14     Params: html string to be used by beautiful soup, baseurl
15     which is passed from commandline
16     Return: Array of urls that end with pdf files
17     """
18     pdfs = []
19     soup = BeautifulSoup(html, 'html.parser')
20     for link in soup.find_all('a', href=True):
21         linkFound = link.get('href')
22         if isAbsolute(linkFound) == False:
23             linkFound = urljoin(baseurl, linkFound)
24
```

```

25         resp = request(linkFound)
26         if resp is not None:
27             contentType = resp.info().type
28             responseCode = resp.getcode()
29
30             if 'application/pdf' in contentType and responseCode
31                 == 200:
32                 finalURL = resp.geturl()
33                 print "Original URI:", linkFound
34                 print "Final URI:", finalURL
35                 # might not contain it
36                 try:
37                     byteSize = resp.headers['content-length']
38                 except:
39                     byteSize = len(resp.read())
40                 print "Bytes: ", byteSize, "\n"
41                 pdfs.append(finalURL)
42
43     return pdfs
44
45 def request(uri):
46     """
47     Params: URI to be requested
48     Return: http get response
49     """
50     try:
51         reqHeaders = {'User-Agent': 'Mozilla 5.10'}
52         req = Request(uri, headers=reqHeaders)
53         response = urlopen(req)
54         return response
55     except (HTTPError, ValueError, URLError) as e:
56         pass
57     except BadStatusLine:
58         # print "**Connection closed early For:**", "\n", uri, "\n"
59         pass
60     except KeyboardInterrupt:
61         print ""
62         exit()
63
64 def isAbsolute(url):
65     """
66     Taken from stackoverflow post
67     """
68     try:

```

```

69         return bool(urlparse(url).netloc)
70     except:
71         return False
72
73
74 if __name__ == "__main__":
75
76     if len(sys.argv) == 2:
77         response = request(sys.argv[1])
78
79         if response is None:
80             print "Initial link can't be bad"
81             print "Must contain http:// or https:// and must be
              reachable"
82             exit()
83
84         pdfs = findPdfs(response.read(), response.geturl())
85     else:
86         print "Usage: python pdfCrawl.py URI"
87         exit()

```

Listing 5: Python script that searches for links that end in pdf files

This script was written in python, and requires version 2.7 which is currently the default for mac computers and ODU CS department's servers. My solution took an iterative approach doing one URI at a time and waiting for each response until moving onto the next URI found. This program takes advantage of the built in libraries:

- sys
- urllib2
- urlparse
- httplib

It also uses the third party library Beautiful Soup to parse html content received using this program.

The script is run like so:

```
python pdfCrawl.py URI
```


Once `pdfCrawl.py` is run it first checks if there is indeed a URI provided via command line arguments. Then it will pass the first argument after the script name to the function `request`, which takes a properly formatted URI and performs an HTTP get request using the `urllib2` library. When performing this request, the `urllib2` library takes into consideration: infinite loops from 300 responses, incorrect formatted URIs, no response code at all, and 400 response codes for client errors [1]. I also included the use of the `httplib` library into this function because there were sometimes special errors when the get request could never fulfill a connection to the server. If none of these errors occurred the request function would return the HTTP get response, otherwise it would return nothing.

After the first request was made it would be passed to `findPdfs` function which would use Beautiful Soup to find all the `html a` elements that contained `href` tags to another URI [2]. I would then iterate through each of the URIs found on the page and request again each of those URIs to determine if the URI would point to pdf file. If the final URI provided a content-type of `application/pdf` and a response code of 200 it was considered a pdf file.

One of the test cases that came up is whether a URI found in the html document was absolute or relative. Using a script provided from a Stackoverflow.com post, I created a function that determined if a string was relative or absolute [5]. If it was relative, it would be merged with the original final URI provided from command line to create an absolute URI. There was one case, found in Listing 8, that actually didn't return content-length, meaning I had to count the bytes from the response's content instead of getting it from the header information. When the `findPdfs` function ends it returns an array of pdfs that can be used for further use.

The URIs I used for this problem were:

- `http://www.cs.odu.edu/~mln/teaching/cs532-s17/test/pdfs.html`
- `http://www.cs.odu.edu/~zeil`
- `http://www.cs.odu.edu/~nadeem/classes/cs752-S11/`

I ran my script and then saved the output to text files, they are as follows:

1	Original URI: <code>http://www.cs.odu.edu/~mln/pubs/ht-2015/hypertext-2015-temporal-violations.pdf</code>
2	Final URI: <code>http://www.cs.odu.edu/~mln/pubs/ht-2015/hypertext-2015-temporal-violations.pdf</code>

3 Bytes: 2184076
 4
 5 Original URI: <http://www.cs.odu.edu/~mln/pubs/tpdl-2015/tpdl-2015-annotations.pdf>
 6 Final URI: <http://www.cs.odu.edu/~mln/pubs/tpdl-2015/tpdl-2015-annotations.pdf>
 7 Bytes: 622981
 8
 9 Original URI: <http://arxiv.org/pdf/1512.06195>
 10 Final URI: <https://arxiv.org/pdf/1512.06195.pdf>
 11 Bytes: 1748961
 12
 13 Original URI: <http://www.cs.odu.edu/~mln/pubs/tpdl-2015/tpdl-2015-off-topic.pdf>
 14 Final URI: <http://www.cs.odu.edu/~mln/pubs/tpdl-2015/tpdl-2015-off-topic.pdf>
 15 Bytes: 4308768
 16
 17 Original URI: <http://www.cs.odu.edu/~mln/pubs/tpdl-2015/tpdl-2015-stories.pdf>
 18 Final URI: <http://www.cs.odu.edu/~mln/pubs/tpdl-2015/tpdl-2015-stories.pdf>
 19 Bytes: 1274604
 20
 21 Original URI: <http://www.cs.odu.edu/~mln/pubs/tpdl-2015/tpdl-2015-profiling.pdf>
 22 Final URI: <http://www.cs.odu.edu/~mln/pubs/tpdl-2015/tpdl-2015-profiling.pdf>
 23 Bytes: 639001
 24
 25 Original URI: <http://www.cs.odu.edu/~mln/pubs/jcdl-2014/jcdl-2014-brunelle-damage.pdf>
 26 Final URI: <http://www.cs.odu.edu/~mln/pubs/jcdl-2014/jcdl-2014-brunelle-damage.pdf>
 27 Bytes: 2205546
 28
 29 Original URI: <http://bit.ly/1ZDatNK>
 30 Final URI: <http://www.cs.odu.edu/~mln/pubs/jcdl-2015/jcdl-2015-temporal-intention.pdf>
 31 Bytes: 720476
 32
 33 Original URI: <http://www.cs.odu.edu/~mln/pubs/jcdl-2015/jcdl-2015-mink.pdf>
 34 Final URI: <http://www.cs.odu.edu/~mln/pubs/jcdl-2015/jcdl-2015-mink.pdf>

```

35 Bytes: 1254605
36
37 Original URI: http://www.cs.odu.edu/~mln/pubs/jcdl-2015/jcdl
    -2015-arabic-sites.pdf
38 Final URI: http://www.cs.odu.edu/~mln/pubs/jcdl-2015/jcdl-2015-
    arabic-sites.pdf
39 Bytes: 709420
40
41 Original URI: http://www.cs.odu.edu/~mln/pubs/jcdl-2015/jcdl
    -2015-dictionary.pdf
42 Final URI: http://www.cs.odu.edu/~mln/pubs/jcdl-2015/jcdl-2015-
    dictionary.pdf
43 Bytes: 2350603

```

Listing 6: Output from <http://www.cs.odu.edu/~mln/teaching/cs532-s17/test/pdfs.html>

```

1 Original URI: http://www.cs.odu.edu/~zeil/vita.pdf
2 Final URI: http://www.cs.odu.edu/~zeil/vita.pdf
3 Bytes: 91987

```

Listing 7: Output from <http://www.cs.odu.edu/~zeil>

```

1 Original URI: http://www.cs.odu.edu/~nadeem/classes/cs752-S11/
    s11/material/Sample_Review_1.pdf
2 Final URI: http://www.cs.odu.edu/~nadeem/classes/cs752-S11/s11/
    material/Sample_Review_1.pdf
3 Bytes: 51693
4
5 Original URI: http://www.cs.odu.edu/~nadeem/classes/cs752-S11/
    s11/material/Lec-01_Course-Introduction.pdf
6 Final URI: http://www.cs.odu.edu/~nadeem/classes/cs752-S11/s11/
    material/Lec-01_Course-Introduction.pdf
7 Bytes: 2647409
8
9 Original URI: http://www.cs.odu.edu/~nadeem/classes/cs752-S11/
    s11/material/Lec-02_PHY-Fundamentals.pdf
10 Final URI: http://www.cs.odu.edu/~nadeem/classes/cs752-S11/s11/
    material/Lec-02_PHY-Fundamentals.pdf
11 Bytes: 1737882
12
13 Original URI: http://www.cs.ucsb.edu/~ebelding/courses/284/s06/
    papers/80211_adhoc.pdf
14 Final URI: http://www.cs.ucsb.edu/~ebelding/courses/284/s06/
    papers/80211_adhoc.pdf

```

15 Bytes: 723511
 16
 17 Original URI: <http://www.cs.odu.edu/~nadeem/classes/cs752-S11/s11/material/Lec-03.MAC-I.pdf>
 18 Final URI: <http://www.cs.odu.edu/~nadeem/classes/cs752-S11/s11/material/Lec-03.MAC-I.pdf>
 19 Bytes: 1624694
 20
 21 Original URI: <http://home.eng.iastate.edu/~daji/papers/ton07.pdf>
 22 Final URI: <http://home.eng.iastate.edu/~daji/papers/ton07.pdf>
 23 Bytes: 1068921
 24
 25 Original URI: <http://research.microsoft.com/en-us/um/people/padmanab/papers/imc2005.pdf>
 26 Final URI: <http://research.microsoft.com/en-us/um/people/padmanab/papers/imc2005.pdf>
 27 Bytes: 193593
 28
 29 Original URI: <http://www.cs.odu.edu/~nadeem/classes/cs752-S11/s11/material/Lec-03.MAC-I.pdf>
 30 Final URI: <http://www.cs.odu.edu/~nadeem/classes/cs752-S11/s11/material/Lec-03.MAC-I.pdf>
 31 Bytes: 1624694
 32
 33 Original URI: <http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=E018BA1D3D65453E8DA2B92041291AC5?doi=10.1.1.10.6560&rep=rep1&type=pdf>
 34 Final URI: <http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=E018BA1D3D65453E8DA2B92041291AC5?doi=10.1.1.10.6560&rep=rep1&type=pdf>
 35 Bytes: 299641
 36
 37 Original URI: <http://www.csc.ncsu.edu/faculty/rhee/export/zmacsensys.pdf>
 38 Final URI: <http://www4.ncsu.edu/~rhee/export/zmacsensys.pdf>
 39 Bytes: 283575
 40
 41 Original URI: <http://h10032.www1.hp.com/ctg/Manual/c00186949.pdf>
 42 Final URI: <http://h10032.www1.hp.com/ctg/Manual/c00186949.pdf>
 43 Bytes: 313323
 44
 45 Original URI: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.3.1887&rep=rep1&type=pdf>
 46 Final URI: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.3.1887&rep=rep1&type=pdf>

47 Bytes: 107248
 48
 49 Original URI: <http://sing.stanford.edu/pubs/sing-10-00.pdf>
 50 Final URI: <http://sing.stanford.edu/pubs/sing-10-00.pdf>
 51 Bytes: 1213250
 52
 53 Original URI: <http://www.cs.odu.edu/~nadeem/classes/cs752-S11/s11/material/Lec-05.mac.CSMACN.pdf>
 54 Final URI: <http://www.cs.odu.edu/~nadeem/classes/cs752-S11/s11/material/Lec-05.mac.CSMACN.pdf>
 55 Bytes: 2725911
 56
 57 Original URI: <http://www.csc.ncsu.edu/faculty/rhee/export/zmacsensys.pdf>
 58 Final URI: <http://www4.ncsu.edu/~rhee/export/zmacsensys.pdf>
 59 Bytes: 283575
 60
 61 Original URI: <http://www.cse.wustl.edu/~lu/papers/sensys07.pdf>
 62 Final URI: <http://www.cse.wustl.edu/~lu/papers/sensys07.pdf>
 63 Bytes: 540100
 64
 65 Original URI: <http://pages.cs.wisc.edu/~suman/courses/838/f06/zigbee-myers-talk.pdf>
 66 Final URI: <http://pages.cs.wisc.edu/~suman/courses/838/f06/zigbee-myers-talk.pdf>
 67 Bytes: 790715
 68
 69 Original URI: <http://www.cs.odu.edu/~nadeem/classes/cs752-S11/s11/material/Moral.pdf>
 70 Final URI: <http://www.cs.odu.edu/~nadeem/classes/cs752-S11/s11/material/Moral.pdf>
 71 Bytes: 418388
 72
 73 Original URI: http://portal.acm.org/ft_gateway.cfm?id=989487&type=pdf&coll=&dl=ACM&CFID=15151515&CFTOKEN=6184618
 74 Final URI: http://delivery.acm.org/10.1145/990000/989487/p222-so.pdf?ip=128.82.17.156&id=989487&acc=ACTIVE%20SERVICE&key=B33240AC40EC9E30%2E9EA977942CF5A36F%2E4D4702B0C3E38B35%2E4D4702B0C3E38B35&CFID=892728580&CFTOKEN=12861955&__acm__=1485275670_70d469714eb0fa91b5245531e7f80f8b
 75 Bytes: 238849
 76
 77 Original URI: http://portal.acm.org/ft_gateway.cfm?id=1140286&type=pdf&coll=&dl=ACM&CFID=15151515&CFTOKEN=6184618
 78 Final URI: <http://delivery.acm.org/10.1145/1150000/1140286/p63->

mishra.pdf?ip=128.82.17.156&id=1140286&acc=ACTIVE%20SERVICE&
 key=B33240AC40EC9E30%2E9EA977942CF5A36F%2E4D4702B0C3E38B35%2
 E4D4702B0C3E38B35&CFID=892728602&CFTOKEN=54630644&__acm__
 =1485275672_d8a18a61b43ec52850bc1038657b09b7
 79 Bytes: 227192
 80
 81 Original URI: http://www.google.com/url?sa=t&source=web&cd=2&ved
 =0CB4QFjAB&url=http%3A%2F%2Fciteseerx.ist.psu.edu%2Fviewdoc%2
 Fdownload%3Bjsessionid%3D6B0972E63A8F1577EFFDB76884E97752%3
 Fdoi%3D10.1.1.12.6578%26rep%3Drep1%26type%3Dpdf&ei=
 _1ZQTbjXEYH-8Aa4xfytDg&usg=AFQjCNGykWNj3G7Rl09eVNHnVk9v5-BRFw
 82 Final URI: http://citeseerx.ist.psu.edu/viewdoc/download;
 jsessionid=6B0972E63A8F1577EFFDB76884E97752?doi
 =10.1.1.12.6578&rep=rep1&type=pdf
 83 Bytes: 203451
 84
 85 Original URI: http://ccr.sigcomm.org/online/files/p135-chandra.
 pdf
 86 Final URI: http://ccr.sigcomm.org/online/files/p135-chandra.pdf
 87 Bytes: 1573870
 88
 89 Original URI: http://portal.acm.org/ft_gateway.cfm?id=1023742&
 type=pdf&coll=&dl=ACM&CFID=15151515&CFTOKEN=6184618
 90 Final URI: http://delivery.acm.org/10.1145/1030000/1023742/p216-
 bahl.pdf?ip=128.82.17.156&id=1023742&acc=ACTIVE%20SERVICE&key
 =B33240AC40EC9E30%2E9EA977942CF5A36F%2E4D4702B0C3E38B35%2
 E4D4702B0C3E38B35&CFID=892728641&CFTOKEN=45286051&__acm__
 =1485275677_8e1147d1848f4693c08792abdaeabfc0
 91 Bytes: 326607
 92
 93 Original URI: http://portal.acm.org/ft_gateway.cfm?id=1147554&
 type=pdf&coll=&dl=ACM&CFID=15151515&CFTOKEN=6184618
 94 Final URI: http://delivery.acm.org/10.1145/1150000/1147554/p301-
 wang.pdf?ip=128.82.17.156&id=1147554&acc=ACTIVE%20SERVICE&key
 =B33240AC40EC9E30%2E9EA977942CF5A36F%2E4D4702B0C3E38B35%2
 E4D4702B0C3E38B35&CFID=892728655&CFTOKEN=39639930&__acm__
 =1485275677_2840e702eac9fe87e953ebc4240b6dc7
 95 Bytes: 909703
 96
 97 Original URI: http://www.cs.odu.edu/~nadeem/classes/cs752-S11/
 s11/material/Lec-07_mac-rate-control.pdf
 98 Final URI: http://www.cs.odu.edu/~nadeem/classes/cs752-S11/s11/
 material/Lec-07_mac-rate-control.pdf
 99 Bytes: 2941244
 100

101 Original URI: <http://www.ee.duke.edu/~romit/pubs/capture-Secon07.pdf>
102 Final URI: <http://people.ee.duke.edu/~romit/pubs/capture-Secon07.pdf>
103 Bytes: 128942
104
105 Original URI: <http://www.ee.duke.edu/~romit/pubs/beamcast.pdf>
106 Final URI: <http://people.ee.duke.edu/~romit/pubs/beamcast.pdf>
107 Bytes: 2204185
108
109 Original URI: http://portal.acm.org/ft_gateway.cfm?id=989483&type=pdf&CFID=7885191&CFTOKEN=48457240
110 Final URI: http://delivery.acm.org/10.1145/990000/989483/p187-zhao.pdf?ip=128.82.17.156&id=989483&acc=ACTIVE%20SERVICE&key=B33240AC40EC9E30%2E9EA977942CF5A36F%2E4D4702B0C3E38B35%2E4D4702B0C3E38B35&CFID=892729367&CFTOKEN=78401715&__acm__=1485275766_50d240888f3c288839e2a76727714b73
111 Bytes: 152155
112
113 Original URI: http://portal.acm.org/ft_gateway.cfm?id=581866&type=pdf&CFID=7885191&CFTOKEN=48457240
114 Final URI: http://delivery.acm.org/10.1145/590000/581866/01026005.pdf?ip=128.82.17.156&id=581866&acc=ACTIVE%20SERVICE&key=B33240AC40EC9E30%2E9EA977942CF5A36F%2E4D4702B0C3E38B35%2E4D4702B0C3E38B35&CFID=892729375&CFTOKEN=14371290&__acm__=1485275766_97433652ed1e974363e177077bfb32a0
115 Bytes: 378490
116
117 Original URI: <http://www.cs.wisc.edu/~suman/pubs/chop.pdf>
118 Final URI: <http://pages.cs.wisc.edu/~suman/pubs/chop.pdf>
119 Bytes: 284764
120
121 Original URI: http://portal.acm.org/ft_gateway.cfm?id=1614353&type=pdf&CFID=7885191&CFTOKEN=48457240
122 Final URI: http://delivery.acm.org/10.1145/1620000/1614353/p297-shrivastava.pdf?ip=128.82.17.156&id=1614353&acc=ACTIVE%20SERVICE&key=B33240AC40EC9E30%2E9EA977942CF5A36F%2E4D4702B0C3E38B35%2E4D4702B0C3E38B35&CFID=892729396&CFTOKEN=69450836&__acm__=1485275769_95c71c48af59882e576e1f66ece75c4d
123 Bytes: 1059205
124
125 Original URI: http://portal.acm.org/ft_gateway.cfm?id=1215981&type=pdf&CFID=7885191&CFTOKEN=48457240
126 Final URI: <http://delivery.acm.org/10.1145/1220000/1215981/p8->

mishra.pdf?ip=128.82.17.156&id=1215981&acc=ACTIVE%20SERVICE&
 key=B33240AC40EC9E30%2E9EA977942CF5A36F%2E4D4702B0C3E38B35%2
 E4D4702B0C3E38B35&CFID=892729409&CFTOKEN=36913017&__acm__
 =1485275770_e9ca53263cc91df992afed3cd70d04fd
 127 Bytes: 259166
 128
 129 Original URI: http://www.cise.ufl.edu/~helmy/papers/Gender-based
 -Udayan-MSWM.pdf
 130 Final URI: http://www.cise.ufl.edu/~helmy/papers/Gender-based-
 Udayan-MSWM.pdf
 131 Bytes: 398787
 132
 133 Original URI: http://db.csail.mit.edu/pubs/mobicom06.pdf
 134 Final URI: http://db.csail.mit.edu/pubs/mobicom06.pdf
 135 Bytes: 1553353
 136
 137 Original URI: http://www.cs.umass.edu/~dganeshan/papers/MobiSys10
 -CrowdSearch.pdf
 138 Final URI: https://people.cs.umass.edu/~dganeshan/papers/
 MobiSys10-CrowdSearch.pdf
 139 Bytes: 725104
 140
 141 Original URI: http://www.winlab.rutgers.edu/~suhas/
 SuhasMathur-Mobisys2010.pdf
 142 Final URI: http://www.winlab.rutgers.edu/~suhas/
 SuhasMathur-Mobisys2010.pdf
 143 Bytes: 1226531
 144
 145 Original URI: http://www.cs.dartmouth.edu/~sensorlab/pubs/
 cenceme_sensys08.pdf
 146 Final URI: http://www.cs.dartmouth.edu/~sensorlab/pubs/
 cenceme_sensys08.pdf
 147 Bytes: 1131508
 148
 149 Original URI: http://db.csail.mit.edu/pubs/mobicom06.pdf
 150 Final URI: http://db.csail.mit.edu/pubs/mobicom06.pdf
 151 Bytes: 1553353
 152
 153 Original URI: http://www.cs.odu.edu/~nadeem/classes/cs752-S11/
 s11/material/Lec-12_crowd-search.pdf
 154 Final URI: http://www.cs.odu.edu/~nadeem/classes/cs752-S11/s11/
 material/Lec-12_crowd-search.pdf
 155 Bytes: 4538638
 156
 157 Original URI: http://portal.acm.org/ft_gateway.cfm?id=1067193&

type=pdf&CFID=9869336&CFTOKEN=84026223
 158 Final URI: http://delivery.acm.org/10.1145/1070000/1067193/p205-
 youssef.pdf?ip=128.82.17.156&id=1067193&acc=ACTIVE%20SERVICE&
 key=B33240AC40EC9E30%2E9EA977942CF5A36F%2E4D4702B0C3E38B35%2
 E4D4702B0C3E38B35&CFID=892729473&CFTOKEN=67511478&__acm__
 =1485275780_f5922e724cd7459b863ac09873d6eba7
 159 Bytes: 382306
 160
 161 Original URI: http://portal.acm.org/ft_gateway.cfm?id=1874078&
 type=pdf&CFID=9869336&CFTOKEN=84026223
 162 Final URI: http://delivery.acm.org/10.1145/1880000/1874078/p787-
 martin.pdf?ip=128.82.17.156&id=1874078&acc=ACTIVE%20SERVICE&
 key=B33240AC40EC9E30%2E9EA977942CF5A36F%2E4D4702B0C3E38B35%2
 E4D4702B0C3E38B35&CFID=892729483&CFTOKEN=80101923&__acm__
 =1485275781_4e369c4ffc00555f20a48b1da1574372
 163 Bytes: 578217
 164
 165 Original URI: http://portal.acm.org/ft_gateway.cfm?id=1023728&
 type=pdf&CFID=9869336&CFTOKEN=84026223
 166 Final URI: http://delivery.acm.org/10.1145/1030000/1023728/p70-
 haeberlen.pdf?ip=128.82.17.156&id=1023728&acc=ACTIVE%20
 SERVICE&key=B33240AC40EC9E30%2E9EA977942CF5A36F%2
 E4D4702B0C3E38B35%2E4D4702B0C3E38B35&CFID=892729566&CFTOKEN
 =65256139&__acm__=1485275792_53807802c6ffe8d3c7cf9bcc9599ee14
 167 Bytes: 658303
 168
 169 Original URI: http://www.cs.odu.edu/~nadeem/classes/cs752-S11/
 s11/material/Lec-13_Horus.pdf
 170 Final URI: http://www.cs.odu.edu/~nadeem/classes/cs752-S11/s11/
 material/Lec-13_Horus.pdf
 171 Bytes: 2016052
 172
 173 Original URI: http://www.cs.ucsb.edu/~ebelding/txt/wmcsa99.pdf
 174 Final URI: http://people.cs.ucsb.edu/ebelding/sites/people/
 ebelding/files/publications/wmcsa99.pdf
 175 Bytes: 224182
 176
 177 Original URI: http://www.eecs.harvard.edu/~htk/publication/2000-
 mobi-karp-kung.pdf
 178 Final URI: http://www.eecs.harvard.edu/~htk/publication/2000-
 mobi-karp-kung.pdf
 179 Bytes: 193263
 180
 181 Original URI: http://www.cs.odu.edu/~nadeem/papers/
 routing_mass_web.pdf

182 Final URI: http://www.cs.odu.edu/~nadeem/papers/routing_mass_web.pdf
183 Bytes: 249519
184
185 Original URI: <http://pdos.csail.mit.edu/papers/grid:mobicom03/paper.pdf>
186 Final URI: <https://pdos.csail.mit.edu/papers/grid:mobicom03/paper.pdf>
187 Bytes: 284493
188
189 Original URI: http://pdos.csail.mit.edu/papers/roofnet:exor-sigcomm05/roofnet_exor-sigcomm05.pdf
190 Final URI: https://pdos.csail.mit.edu/papers/roofnet:exor-sigcomm05/roofnet_exor-sigcomm05.pdf
191 Bytes: 206781
192
193 Original URI: http://www.ieee-infocom.org/2004/Papers/39_3.PDF
194 Final URI: http://infocom2004.ieee-infocom.org/Papers/39_3.PDF
195 Bytes: 212041
196
197 Original URI: http://www.ieee-infocom.org/2003/papers/43_01.PDF
198 Final URI: http://infocom2003.ieee-infocom.org/papers/43_01.PDF
199 Bytes: 206229
200
201 Original URI: <http://www.ee.duke.edu/~romit/pubs/xrtp.pdf>
202 Final URI: <http://people.ee.duke.edu/~romit/pubs/xrtp.pdf>
203 Bytes: 167025
204
205 Original URI: http://www.cs.odu.edu/~nadeem/classes/cs752-S11/s11/material/Lec-19_Blocked_switched.pdf
206 Final URI: http://www.cs.odu.edu/~nadeem/classes/cs752-S11/s11/material/Lec-19_Blocked_switched.pdf
207 Bytes: 776011
208
209 Original URI: <http://www.winlab.rutgers.edu/~gruteser/papers/ccs308-baik.pdf>
210 Final URI: <http://www.winlab.rutgers.edu/~gruteser/papers/ccs308-baik.pdf>
211 Bytes: 3052631
212
213 Original URI: <http://www.isaac.cs.berkeley.edu/isaac/mobicom.pdf>
214 Final URI: <http://www.isaac.cs.berkeley.edu/isaac/mobicom.pdf>
215 Bytes: 101296
216
217 Original URI: http://www.cs.odu.edu/~nadeem/papers/reputation_TC

```

218 .pdf
218 Final URI: http://www.cs.odu.edu/~nadeem/papers/reputation_TC.
      pdf
219 Bytes: 2928938
220
221 Original URI: http://www.cs.umd.edu/~waa/pubs/wireless-comm-no-
      clothes.pdf
222 Final URI: http://www.cs.umd.edu/~waa/pubs/wireless-comm-no-
      clothes.pdf
223 Bytes: 103857
224
225 Original URI: http://www.cs.utexas.edu/~lili/papers/pub/
      mobicom2004.pdf
226 Final URI: http://www.cs.utexas.edu/~lili/papers/pub/mobicom2004
      .pdf
227 Bytes: 264612
228
229 Original URI: http://www.cs.odu.edu/~nadeem/classes/cs752-S11/
      s11/material/Lec-21_Routing_Misbehavior.pdf
230 Final URI: http://www.cs.odu.edu/~nadeem/classes/cs752-S11/s11/
      material/Lec-21_Routing_Misbehavior.pdf
231 Bytes: 263518
232
233 Original URI: http://www.cs.wisc.edu/~suman/pubs/paradis.pdf
234 Final URI: http://pages.cs.wisc.edu/~suman/pubs/paradis.pdf
235 Bytes: 560178
236
237 Original URI: http://www.cs.dartmouth.edu/~dfk/papers/bratus-
      fingerprint.pdf
238 Final URI: http://www.cs.dartmouth.edu/~dfk/papers/bratus-
      fingerprint.pdf
239 Bytes: 283479

```

Listing 8: Output from <http://www.cs.odu.edu/~nadeem/classes/cs752-S11/>

3

Question

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:
```

Answer

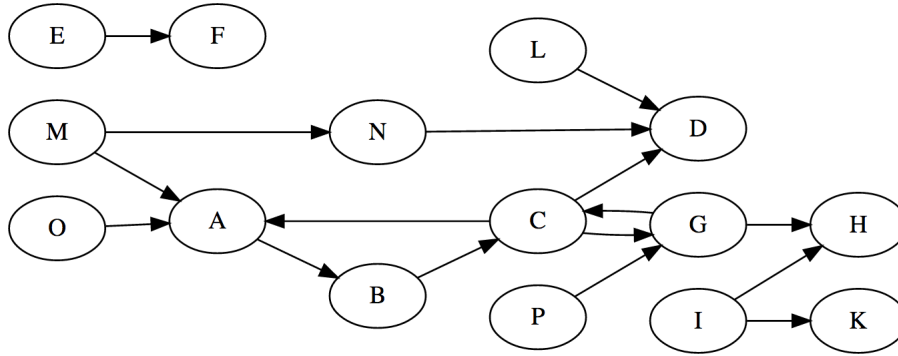


Figure 3: Graph representation generated with WebGraphviz [6]

IN: M, O, P

These values are considered the *IN* values due to the fact that they can reach values that are considered to be in the *SCC* and also because they can't be reached from the *SCC* [4].

SCC: A, B, C, G

These values are considered the *SCC* values because they are at the “heart of the graph.” They either are all nodes that can reach another node along directed links. This can consist of links from the outside in, nodes inside the *SCC* pointing to other nodes inside, or nodes point from the inside out [4].

OUT: D, H

These values are part of the *OUT* because they are accessible from the *SCC* but they cannot link back into it [4].

Tendrils: I, K, L

These values don't reference the *SCC* at any point, but do have links to the *OUT* nodes and therefore they are considered the *tendrils* [4].

Tubes: N

This value isn't part of the “heart of the graph” but it does connect an *IN* node to an *OUT* node in one step, not touching the *SCC* in the process [4].

Disconnected: E, F

These two values are as their title describes - disconnected. They aren't part of the *SCC* and don't connect to anything else on the graph.

References

- [1] "20.6. Urllib2 ? Extensible Library for Opening URLs." 20.6. Urllib2 - Extensible Library for Opening URLs - Python 2.7.13 Documentation. Python Software Foundation, n.d. Web. 24 Jan. 2017. <https://docs.python.org/2/library/urllib2.html>.
- [2] Richardson, Leonard. "Beautiful Soup Documentation." Beautiful Soup Documentation - Beautiful Soup 4.4.0 Documentation. N.p., n.d. Web. 24 Jan. 2017. <https://www.crummy.com/software/BeautifulSoup/bs4/doc/>.
- [3] Stenberg, Daniel. "Curl.1 the Man Page." Curl - How To Use. N.p., n.d. Web. 24 Jan. 2017. <https://curl.haxx.se/docs/manpage.html>.
- [4] Broder, Andrei, Ravi Kumar, Farzin Maghoul, Prabhakar Raghavan, Sridhar Rajagopalan, Raymie Stata, Andrew Tomkins, and Janet Wiener. "Graph Structure in the Web." 9th International World Wide Web Conference, June 2000. Web. 24 Jan. 2017. <http://www9.org/w9cdrom/160/160.html>.
- [5] Lalinsk, Luk. "How Can I Check If a URL Is Absolute Using Python?" Stack Overflow. N.p., n.d. Web. 24 Jan. 2017. <http://stackoverflow.com/questions/8357098/how-can-i-check-if-a-url-is-absolute-using-python>.
- [6] "Webgraphviz." Webgraphviz. N.p., n.d. Web. 24 Jan. 2017. <http://www.webgraphviz.com/>.