


FH C S 021A 10479 RK SU15 FH C S 030B 41756 HA SP15 FH C S 001B 31129 EH W15 FH C S 022A 31242 RK W15

ASSIGNMENTS, TESTS AND SURVEYS

Working on Assignment

Assignment 5

(worth 140 points)

Your task is to implement a simple web crawler. A web crawler is a program that systematically browses the web to extract information.

Start with the module provided under Resources `crawler.py`. This is more than a template: it contains some **functions that have been written for you: `crawl` and `ok_to_crawl`.** Do not change these functions.

There are 3 places in `crawler.py` where you will find **# TO DO comments**.

These identify the functions that you have to define.

1. You'll need to write the **`main` function**. The task has been outlined for you. You may use additional functions, for example to print out the urls, etc...
2. You'll need to fill in the code for **`get_page`**. Follow the outline provided. Note that you may have different kinds of errors generated in `get_page`. Make sure you handle both `URLError` and `DecodeError` and generate the appropriate message for each.
3. You'll need to write **`extract_links`**. Follow the outline provided. The function **`ok_to_crawl`** has been provided for you. You'll have to call it from `extract_links`.
4. `crawl` and `ok_to_crawl` have been provided for you. Because we don't want to overwhelm any particular site, we are limiting ourselves to crawling **no more than 10 urls**. The `crawl` function takes care of that limit while `ok_to_crawl` ensures that we are implementing polite crawling. It also filters `mailto:` links and `javascript:` links. **`ok_to_crawl` takes an absolute url**. If it is given a relative url, it generates an error message and returns `False`. **Make sure you check the error messages.**

You need to be able to **invoke your crawler from the command line** by typing:

```
python crawler.py somewebaddress
somewebaddress will be your seed url.
```

Make sure you **read module section 17.7** for some important information about web crawlers.

Testing:

Make sure you test your module before submitting it!

For simple initial testing you should use a small set of web pages available

at <http://googledrive.com/host/0BwBC7CTQMPGqfJJaRWMYTTIVRUpjclBPWnktcU5pYnA4ek9WQUxFOXVDWm16RURTVVBbWM/CS21Ahome.html>.

Your seed url will now be:

<http://googledrive.com/host/0BwBC7CTQMPGqfJJaRWMYTTIVRUpjclBPWnktcU5pYnA4ek9WQUxFOXVDWm16RURTVVBbWM/CS21Ahome.html>

To invoke your crawler you'll type (or copy and paste) the following in the Terminal window (on a single line):

```
python crawler.py
http://googledrive.com/host/0BwBC7CTQMPGqfJJaRWMYTTIVRUpjclBPWnktcU5pYnA4ek9WQUxFOXVDWm16RURTVVBbWM/CS21Ahome.html
```

Your **output file `crawled.txt`** corresponding to this seed should contain the 6 absolute urls:

```
http://googledrive.com/host/0BwBC7CTQMPGqfJJaRWMYTTIVRUpjclBPWnktcU5pYnA4ek9WQUxFOXVDWm16RURTVVBbWM/CS21Ahome.html
http://googledrive.com/host/0BwBC7CTQMPGqfJJaRWMYTTIVRUpjclBPWnktcU5pYnA4ek9WQUxFOXVDWm16RURTVVBbWM/homework.html
http://googledrive.com/host/0BwBC7CTQMPGqfJJaRWMYTTIVRUpjclBPWnktcU5pYnA4ek9WQUxFOXVDWm16RURTVVBbWM/assignments.html
http://googledrive.com/host/0BwBC7CTQMPGqfJJaRWMYTTIVRUpjclBPWnktcU5pYnA4ek9WQUxFOXVDWm16RURTVVBbWM/syllabus.html
http://googledrive.com/host/0BwBC7CTQMPGqfJJaRWMYTTIVRUpjclBPWnktcU5pYnA4ek9WQUxFOXVDWm16RURTVVBbWM/contact.html
http://googledrive.com/host/0BwBC7CTQMPGqfJJaRWMYTTIVRUpjclBPWnktcU5pYnA4ek9WQUxFOXVDWm16RURTVVBbWM/lectures.html
```

The order is not relevant.

Once your program works on this url and produces the correct output, you can try it on any seed url. Note that with any seed url, your **output file should not include more than 10 crawled urls**.

Error Checking Requirement:

The program gets the seed url from the command line arguments. If the seed url is missing, the program should print a 'usage' message and no crawling should take place.

For example, if the user types:

```
>python crawler.py
```

A message such as the following should be printed:

```
Usage: crawler.py seed_url
```

Home
CourseMap
Announcements
Syllabus
Modules
Assignments, Tests and Surveys
Discussion and Private Messages
Resources
Gradebook
Python Documentation
Learning with Python 3
Dive Into Python 3
Python Style Guide
Docstring Conventions
The Python Standard Library
Pythex
Tkinter Reference
Opportunities for CS Students

users present:

Justin Cheng
Shane Duffy
David Gudeman
Rula Khayrallah
Joshua Kurniawan
Jay Lu
Nicole Segovia

Similarly, if the user specifies too many arguments as in:

```
> python crawler.py http://www.foothill.edu/index.php https://www.deanza.edu/
```

An error message should be printed as well and no crawling should take place.

Grading Rubric:

main – 45 points total

- Getting the seed url from the command line arguments
- Printing an error message if the command line arguments provided are too few or too many.
- **Writing all the urls in `crawl_path` to a file called `crawled.txt`** in the working directory. Make sure you print one url per line

get_page – 40 points

- The `with` construct is used to open the url.
- The function handles errors encountered while opening the url and errors due to decoding the page.
- An appropriate message **identifying the url and the error** is generated when an error is encountered.
- A string is returned in all cases.

extract_links – 50 points


- The regular expression extracts all 6 links from the **test document in section 17.7** (also provided under Resources) and it does not extract any additional strings such as "notaur.css" or class="important". Make sure you **test your regular expression in Pythex with the test document** before using it in the crawler.
- The function returns the **set** of urls found on the page that are **OK to crawl** after converting them to **absolute urls**.

Overall style and documentation: 5 points

Start early, ask questions and have fun!

Find and select the file from your computer and then click on "Upload" to upload it.

No file chosen

 [Instructions](#)

| [Gateway](#) | [Etudes](#) | [Conduct Policy](#) | [Privacy Statement](#) |

© 2008 - 2015 Etudes, Inc. All rights reserved.
Etudes20150626 - hades