University of Washington Bothell

CSS 490: Cloud Computing

Program 1: Crawl

Purpose

The programmable Web and cloud is built on HTTP. In this lab we will utilize HTTP programming to "GET" and crawl parts of the HTML based Web. The goal is to familiarize students with HTTP and programmatically using it to access the web. Of course, the majority of the access to the Cloud is done using APIs and development environments which target specific services, these are built on HTTP.

Problem Statement

Create an application which takes two arguments:

1) A URL as a starting point
2) The number of hops from that URL (NumHops)

Your application will download the html from the starting URL which is provided as the first argument to the program. It will parse the html finding the first <a href > reference to other absolute URLs, for instance https://www.w3schools.com/tags/att_a_href.asp . Make sure that you have not previously visited this page (if you have then skip and find the next reference). The application will then download the html from that page and repeat the operation. You will do this NumHops times. Your app will print out to the console the value of the final URL you landed on as well as the html. If you encounter a page without any embedded references you should stop there and print out the result.

Problem Statement Details

- **Example:**

  MyWebCrawl.exe http://courses.washington.edu/css502/dimpsey 5

  Prints out to the console the URL and the html from the URL 5 hops from http://courses.washington.edu/css502/dimpsey

Details

- You may build you app either with C# or Java. For this exercise do not use Python.
- Please make sure to factor your application appropriately.
- Make sure that your program takes in two arguments—do not query the user for the URL or number of hops.

- Your program should gracefully handle all input, including bad input.
- Your program should gracefully handle unreachable sites.
- You should appropriately handle HTTP requests with return codes in the 300s or 400s
- JSoup or similar libraries should not be used for this assignment; you should be making HTTP requests directly.
- Building your program should be easy to do by the grader using your Makefile. Part of the grade will be the ease of creating your executable.
- **An IDE should not be required to build the application. If one is required we may not be able to build it. Even we are able to build it then points will be deducted.**
- You can submit a .class or .jar file for you executable (if using java).
- This should be a command line executable. There should be NO gui interfaces.

Turn In

A .**zip file** which the module named:

- Executable of application
- All code and clear instructions or Makefile on how to build and run the application