**MCS 253P Lab 2**

1. Write a program to parse URLs into components. Input, from stdin, will be a list of URLs, one per line. <http://en.wikipedia.org/wiki/Uniform_resource_locator> Output should be a variable-length list of components which represent the parts of the URL. Print a blank line after each URL. A URL may have the syntax:

scheme://domain:port/path?query\_string#fragment\_id

And many of these parts are optional. Only given fields should be listed in the output for this URL. Output should follow the following form (bold is a literal string you will print and the non-bold in angle brackets will be whatever string you found for that part in the URL):

**scheme:** <scheme>

**domain:** <domain>

**port:** <port>

**path:** <path>

**query\_string:** <query>

**fragment\_id:** <fragment>

<blank line>

Here are some more samples for your testing

https://apis.google.com/js/client.js

http://www.facebook.com/pages/Learn-the-Net/330002341216

ftp://rtfm.mit.edu/pub/

https://www.google.com/search?q=ebay

www.learnthenet.com/web-at-a-glance/url-examples/

http://my.ebay.com:80/ws/eBayISAPI.dll?MyEbayBeta&MyEbay=&gbh=1&guest=1#myfragment

Sample output for the last two URLs above:

**domain:** www.learnthenet.com

**path:** web-at-a-glance/url-examples/

<blank line>

**scheme:** http

**domain:** my.ebay.com

**port:** 80

**path:** ws/eBayISAPI.dll

**query\_string:** MyEbayBeta&MyEbay=&gbh=1&guest=1

**fragment\_id:** myFragment

<blank line>

1. Write a program to count the frequency of occurance of every character in the input. The printable characters in ASCII are from space ‘ ‘ to tilde ‘~’ inclusive.(20-126) Output, to stdout, will be a list of pairs from space through tilde with an integer after the character indicating how many times it appeared in the input. Then print how many whitespace characters (space, tab, newline) then how many other characters (non-printable non-whitespace). Note each space character will be counted twice as a space and as whitespace. Only print the ones that have a count greater than zero (so the number of lines will vary and will only report counts for characters that appeared in the input). Use the man page for bash for your test input. Here is a sample output,

‘ ‘ 24

‘!’ 3

‘“‘ 21

…

‘}’ 14

‘~’ 2

**whitespace characters** 123

**other characters** 12

1. Write a program to convert Roman Numerals into equivalent Decimal Numerals. Input, from stdin, will be a list of Roman Numerals one per line. Output will be a list of pairs: the roman numeral an arrow then the decimal numberal equivalent of the Roman Numeral in the input.

<http://sizes.com/numbers/roman_numerals.htm>

Input example:

II

IV

VIII

XVI

XXXII

LXIV

CXXVIII

CCLVI

DXII

MXXIV

MMXLVIII

MMMMXCVI

MMMMMMMMCXCII

Output example:

II => 2

IV => 4

VIII => 8

XVI => 16

XXXII => 32

LXIV => 64

CXXVIII => 128

CCLVI => 256

DXII => 512

MXXIV => 1024

MMXLVIII => 2048

MMMMXCVI => 4096

MMMMMMMMCXCII => 8192