SCHOOL OF DIGITAL MEDIA AND INFOCOMM TECHNOLOGY
DIPLOMA IN INFOCOMM SECURITY MANAGEMENT

Year 2

ST2614 PROGRAMMING USING PERL AND C

~ ASSIGNMENT 1 ~

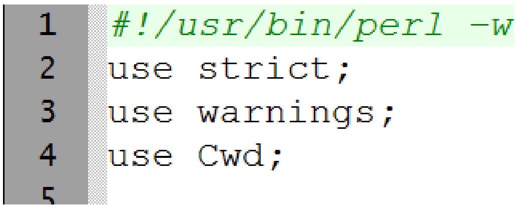
For this assignment you will be implementing a PERL application that does web scraping in order to generate a list of email addresses for further processing. Your application should be able to generate different type of reports, either as text printed directly on the screen, or in the form of html printed directly on the screen.

1. This is an individual assignment.

2. Your application must be submitted online to blackboard before the **17th of May 2013 05:00PM**. Please ensure the submission contains the required PERL file to run the application. Your PERL file should be named **assignment1-{student ID}.pl**.

   Important: Your application <u>must</u> be able to run using Perl 5.10.x interpreter.

3. Do <u>not</u> make use of any additional Modules other than the standard modules *strict, warnings* and *Cwd*.

   ```
   #!/usr/bin/perl -w
   use strict;
   use warnings;
   use Cwd;
   ```

4. On top of the source code add in comments your name, student ID and class.

5. Do <u>not</u> compress your files (that is do **not** zip or rar your files)

6. You must demonstrate your application to the lecturer during the first practical session after the assignment due date.

7. During the demonstration you may be asked to explain on the code that you have produced. Take note, your application may be tested with a fresh data set.

Take note: Points will be subtracted for late submissions. Work that has been copied from others will be awarded zero marks. Students that allow their work to be copied by others will also be awarded zero marks.

1. Background

The student will develop a PERL web scraping application for email from web pages for further processing. The below definition for web scraping are extracted from Wikipedia (http://en.wikipedia.org/wiki/Web_scraping):

> Web scraping (web harvesting or web data extraction) is a computer software technique of extracting information from websites. Usually, such software programs simulate human exploration of the World Wide Web by either implementing low-level Hypertext Transfer Protocol (HTTP), or embedding a fully-fledged web browser, such as Internet Explorer or Mozilla Firefox.
>
> Web scraping is closely related to web indexing, which indexes information on the web using a bot or web crawler and is a universal technique adopted by most search engines. In contrast, web scraping focuses more on the transformation of unstructured data on the web, typically in HTML format, into structured data that can be stored and analyzed in a central local database or spreadsheet. Web scraping is also related to web automation, which simulates human browsing using computer software. Uses of web scraping include online price comparison, weather data monitoring, website change detection, research, web mashup and web data integration.

2. Application feature specifications

- Your application must be able to support the below switches

```
Usage: {file name} [switches]
   -dpath   directory to search for html files (default: .)
   -f[th]   output in text or html mode (default: text)
   -h       print this message and exit
   -s[adn]  sort in ascending, descending or none (default: none)
   -v       print version and exit
```

- The application will search html files within specified folder path by the -dpath as the top level folder. The application will recursively search any child folder(s) that is present in the top level folder and its child folder(s). The default search path will be the present working folder that the script is being run frWom.

- The application will output the data in html format if -fh is specified. Text format will be used by default if none are specified and when -ft is being specified.

- The application will print the help message as shown in the output above with all the possible available switches if the -h is specified.

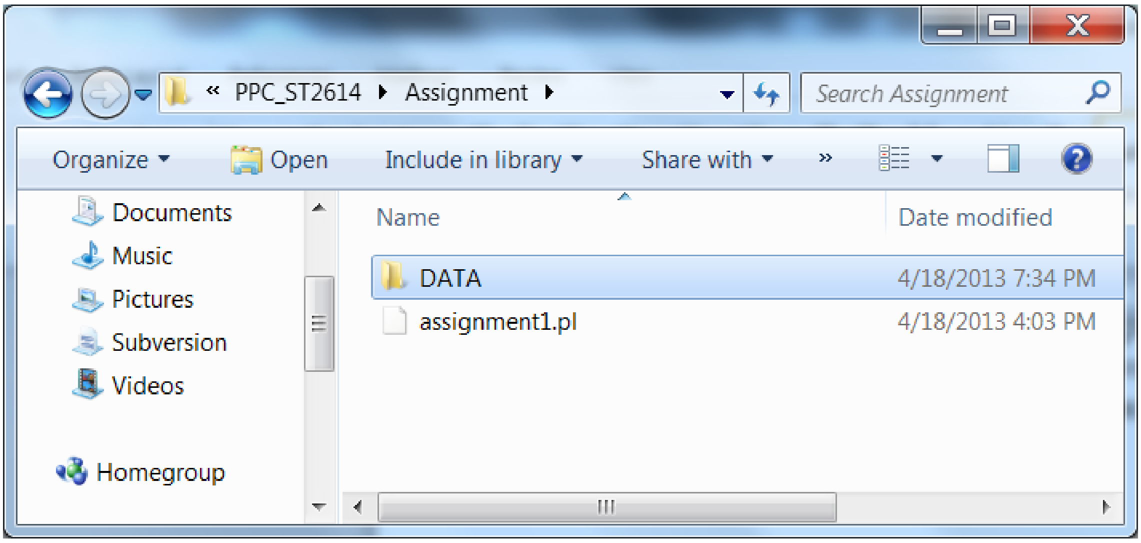
- The application will sort the output data in ascending order if -sa is specified. Descending order will be used if -sd is specified. Output data will not be sorted if none are specified and when -sn is being specified.

- Your application must print the application version number, your name, student ID and class as shown in below example when -v is being specified:

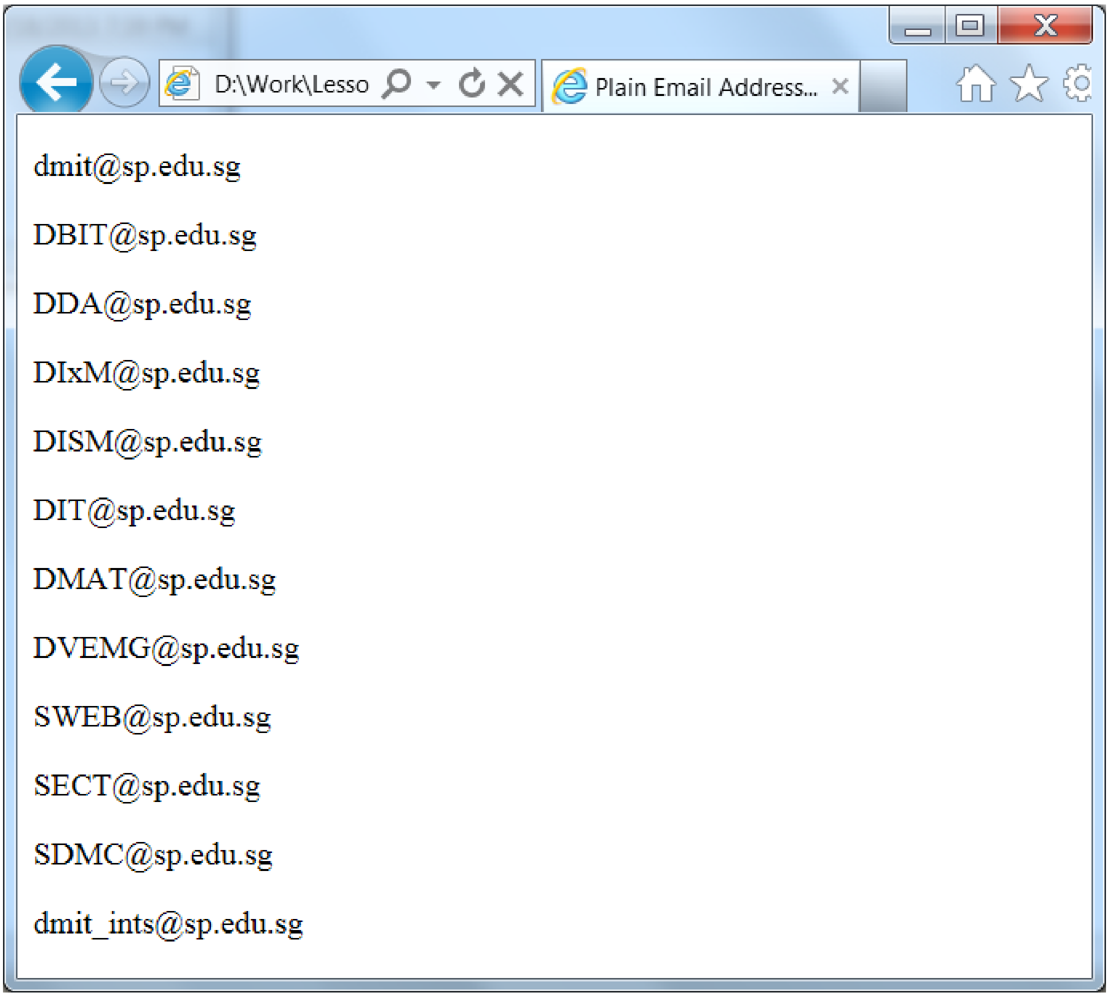
```
*******************************************************************
ST2614 Assignment 1, Ver. 1.0 done by John Tan p1234567 class 2B/10
*******************************************************************
```

3. Data file specifications

- Your application must be able to process data files that are stored in a folder specified by the -dpath switch. You will be given an example DATA folder to practice with. Use –dDATA to process the data files within the DATA folder.



- Below is the simplified version of the HTML file containing the email addresses



dmit@sp.edu.sg

DBIT@sp.edu.sg

DDA@sp.edu.sg

DIxM@sp.edu.sg

DISM@sp.edu.sg

DIT@sp.edu.sg

DMAT@sp.edu.sg

DVEMG@sp.edu.sg

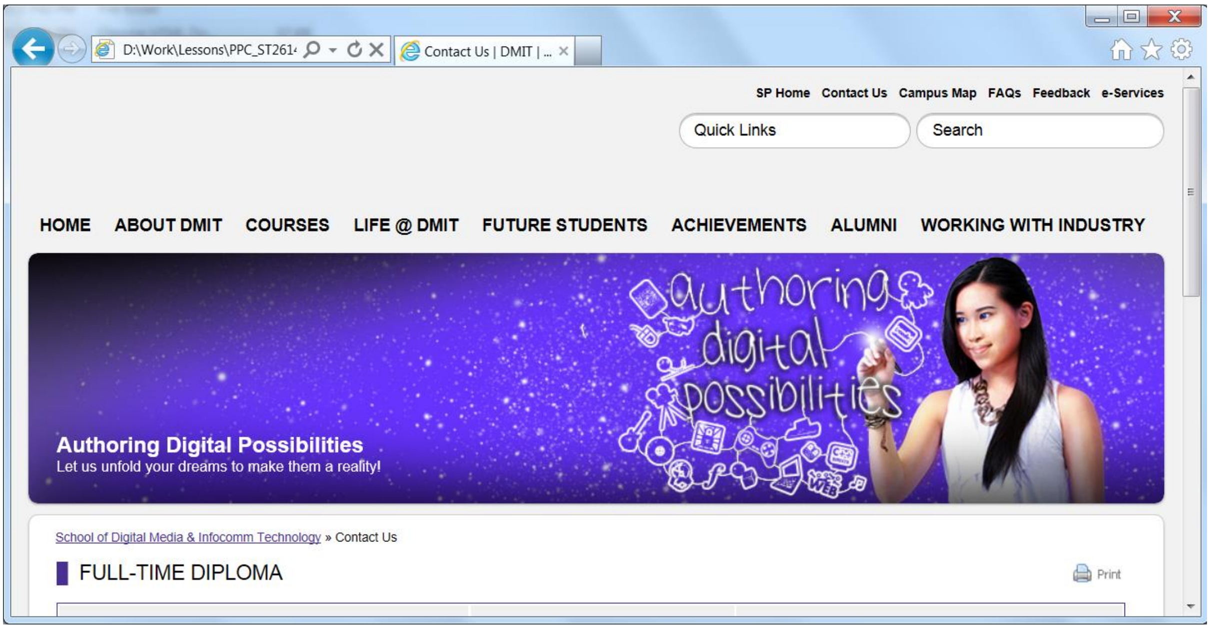
SWEB@sp.edu.sg

SECT@sp.edu.sg

SDMC@sp.edu.sg

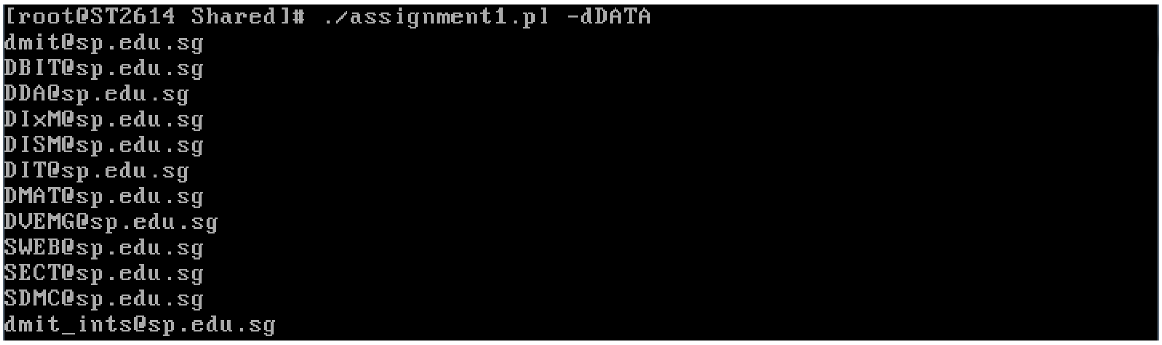
dmit_ints@sp.edu.sg

- Below is the full version of the HTML file containing the email addresses.



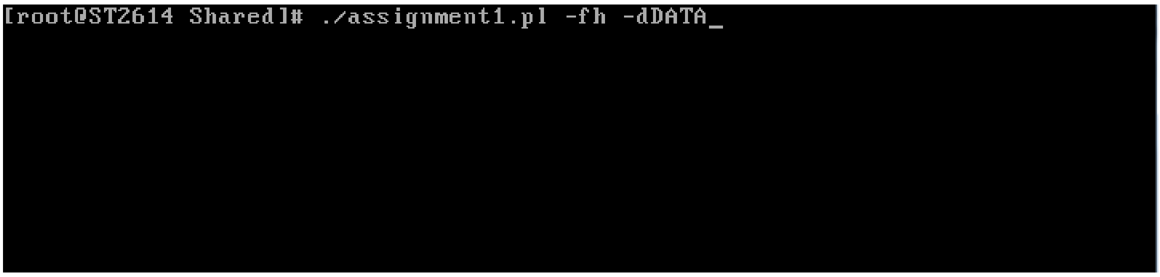
4. Output formats

- If the user chooses to generate a text report, your application must display the list of email addresses onto the screen in the same manner as shown below:



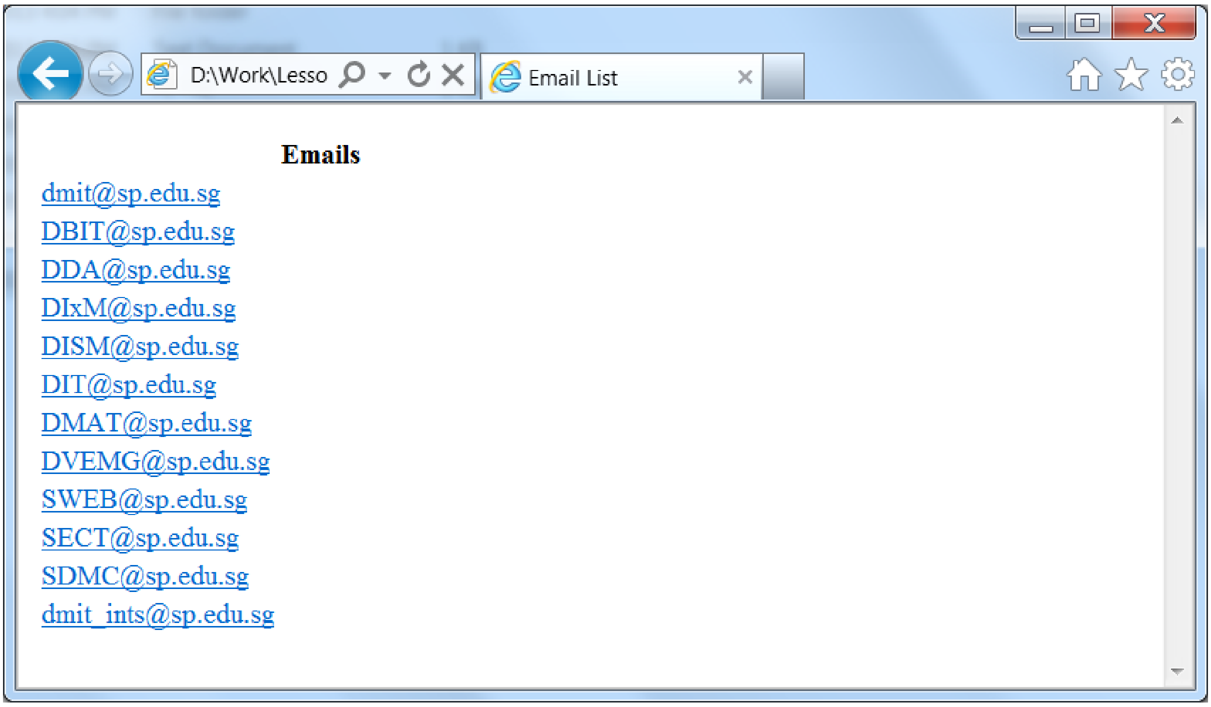
```
[root@ST2614 Shared]# ./assignment1.pl -dDATA
dmit@sp.edu.sg
DBIT@sp.edu.sg
DDA@sp.edu.sg
DIxM@sp.edu.sg
DISM@sp.edu.sg
DIT@sp.edu.sg
DMAT@sp.edu.sg
DVEMG@sp.edu.sg
SWEB@sp.edu.sg
SECT@sp.edu.sg
SDMC@sp.edu.sg
dmit_ints@sp.edu.sg
```

- If the user chooses to generate a HTML formatted report in the manner as shown below:



```
[root@ST2614 Shared]# ./assignment1.pl -fh -dDATA_
```

- The HTML formatted report can be redirected into a file and view from the web browser as shown below:



Emails

dmit@sp.edu.sg
DBIT@sp.edu.sg
DDA@sp.edu.sg
DIxM@sp.edu.sg
DISM@sp.edu.sg
DIT@sp.edu.sg
DMAT@sp.edu.sg
DVEMG@sp.edu.sg
SWEB@sp.edu.sg
SECT@sp.edu.sg
SDMC@sp.edu.sg
dmit_ints@sp.edu.sg

Note: The user will be able to click on the links in the web page to email directly.

5. Bonus features (Optional)

- The application will detect duplicate email addresses and display only one copy of the email address.
- The functionality to search for files within the top level folder and its child folder(s) are implemented using recursion.