Scripting & Computer Environments (CSE 505)

Assignment 2 - Regular Expressions

Deadline: 31st August 2013 @ 9pm

Instructions:

- *Create a file* `<YourRollNumber>.txt`*. This file will contain the command(s) for each of the following questions,* ***along with the explanation of your approach/logic for each****.*
- *A file named* `<YourRollNumber>.script` *generated using the* `script` *command also must be included.*
- *Further upload instructions will be mailed to you.*

1. Write a regular expression to extract only the valid email addresses that match any of the following domains: {com, ac.in, edu, org}. See the accompanying `1.txt` file for some valid and invalid email IDs.
2. Matching IPv4 addresses:
   (a) Write a regex to extract all numbers of the format `W.X.Y.Z` from `2.txt`. Each placeholder can assume any value.
   e.g. `10.2.40.95, 555.77.999.0`
   (b) Now, extract only those numbers of the form `A.A.A.A`.
   e.g. `10.10.10.10, 455.455.455.455`
   (c) This time, extract only those numbers which are between 0 and 255 inclusive.
   e.g. `10.2.90.100, 195.32.0.255`
3. Display only the lines from `3.txt` which satisfy each of the following conditions:
   (a) First character matches with the last character.
   (b) Second character matches with the second last character.
4. Display only those lines in `file 4.txt` that contain a valid HTML heading tag syntax (H1 through H6). A tag is valid if the opening and closing tags are the same. Comparison should be case insensitive.

```
<H1> This is valid heading </H1> and
<H2> This is invalid </H3>
```

5. Filter the following from `5.txt`:

   (a) The lines having comments. Comments start with hash(`#`), single quote (`'`) or double forward slashes (`//`).

   (b) Now, extract all the lines that do not have comments.

6. Write a command to extract only the lines that contain valid vehicle numbers from the file `6.txt`. Valid numbers are 2 letters, a space, 2 digits, 1 or 2 letters, a space and finally a 1,2,3 or 4 digit number.

   e.g. {`AP 07 CJ 12, MH 15 BH 6549, MH 12 CJ 2339, UP 53 R 788`}

7. Replace all the contiguous repeating characters from `7.txt` by a single character. You should handle multiple spaces as well.

   e.g. `'aaabccc'` → `'abc'`

8. Using both `find` and `grep`, list all files in your home directory that were changed less than 10 hours ago, but leave out directories.

9. Using both `sed` and `awk`, achieve the following:

   (a) Delete the first 3 lines of `9.txt`

   (b) Print the lines containing the pattern "an" from the file `10.txt`.

   (c) Substitute all lines in `11.txt` that begin with /* and end with */ by an empty line.

10. Output the lines in the file `12.txt` that match valid floating point and positive integer numbers.

11. Write an `awk` command to rename all the files in the current directory with size greater than 1 MB with the postfix "_big".

    e.g. `file.txt` → `file_big.txt` and `xyz` → `xyz_big`

12. Given the data tabulated in the file `employee.txt`, use the appropriate command(s) to do the following:

    (a) Prepare an inventory report that contains the `Name`, `Rate`, `Hours_Worked` and `NetPay` for each employee. Include line numbers, appropriate report beginning and ending messages too.

    (b) Which employee(s) was paid the most and least amounts respectively?

    (c) For all employees who worked overtime (`>` 10 hrs), add an appraisal amount of 10%.