Assignment 2: sed

CS3423 - Systems Programming — Sam Silvestro

For this assignment, you will use **sed**, **bash**, and the other utilities you have used in class to create a program for use by a municipality for both redacting sensitive information from internal communications, as well as simplifying and standardizing the format of these documents prior to their release and circulation. Your program should take the names of one or more files that are to be redacted as command line arguments.

This assignment requires only sed, bash, and the other utilities used so far in class. **Do not** use awk, Python, or any other languages/utilities.

Redaction and Substitution Rules

For all files specified, the following changes should be made *in place*. No other changes should be made to the file.

- **Driver's License numbers** begin with *xx*`DL`, where *xx* is a two-letter state code identifying the origin of the issuing state. Following this code is a space character, then a license number composed of *at least* 6 digits in length. For example, the following are all valid driver's license numbers:
  - `TXDL 12345678`
  - `VADL 123456`
  - `WADL 1234567890`

  These numbers should be redacted by simply replacing the license number with a sequence of six `X` characters.

- **Credit Card Numbers** Credit card numbers must be redacted, but it is desirable that the censored text still retain information suitable for the identification of both the type of card (i.e., the issuer), as well as its last four digits. Each card network specifies a unique first digit to the cards they issue, and typically contain 16 digits in total. American Express is an exception to this, in which the second number can only be a 4 or a 7, and the total number of digits is only 15. In summary:
  - Visa cards: Begin with a 4 and have 16 digits
  - Master Cards: Begin with a 5 and have 16 digits
  - American Express cards: Begin with a 3, followed by a 4 or a 7, and have 15 digits
  - Discover cards: Begin with a 6 and have 16 digits

  Card number data should be redacted as in the following examples. Note that 16 digit numbers may or may not be separated into groups of four using hyphens – this is optional. Hyphenation of American Express cards into sections of 4, 6, and 5 digits is similarly common, and also optional. Examples of the expected substitutions for each card type are as follows:

  - `5441-4839-9284-3129` → `MC-3129`
  - `3770-123456-78900` → `AMEX-8900`
  - `6093-2033-0662-5389` → `DISC-5389`
  - `4291723799801302` → `VISA-1302`

- **Texas Vehicle License Plate numbers** should be similarly obliterated. Texas vehicle plates appear in one of two formats, but both will be written with the letters `TX` and an optional space preceding them. The first type is six alphanumeric characters, optionally separated by a hyphen in the middle. The second type begins with three alphabetic characters, followed by four digits, again optionally separated by a hyphen. Examples of valid type one Texas license plate numbers are:
  - `TX 32P9ZP`
  - `TX 32P-9ZP`
  - `TX32P9ZP`
  - `TX32P-9ZP`

  Examples of valid type two Texas license plate numbers are:
  - `TX JTK8791`
  - `TX JTK-8791`
  - `TXJTK8791`
  - `TXJTK-8791`

  These numbers should be redacted by simply replacing the license plate number with a sequence of six `X` characters.

- **Current Date Placeholder** The document authors may use the shorthand symbol `<date>` in order to insert the current date (i.e., today's date). Regardless of the date on which your script is run, this placeholder should be updated with the correct current date.

- **Municipality Name Placeholder** The authors of these documents may use the shorthand symbol `<orgname>` in order to designate the full name of their municipality. Any such references should be replaced with the full name: `City of Gainsville, Florida`.

Example

Original redactme.txt:

```
<orgname>
Date: <date>
Title: Memorandum #139

```

```
Dear staff:

Memorandum #139 has been amended as follows, in accordance with the
updated employee operations and purchasing policy*:

The only employees authorized to operate vehicle #102 (license plate
TX JTK8791), vehicle #162 (license plate TX 32P-9ZP), and vehicle #262
(license plate TX AJC-6244) are those employees who possess the following
driver's licenses:

    TXDL 02851332
    TXDL 00748892
    VADL 590401
    FLDL 104281332

Further, usage of city credit cards will be strictly limited to the
following departmental cards, until further notice:

    5441-4839-9284-3129
    3770-123456-78900
    6093-2033-0662-5389
    4291723799801302

Thank you,
   Mgmt

* Policy revision date 2/1/13 (originally passed 7/13/92).
```

Redacted Version of redactme.txt:

```
City of Gainsville, Florida
Date: 09/19/2019
Title: Memorandum #139

Dear staff:

Memorandum #139 has been amended as follows, in accordance with the
updated employee operations and purchasing policy*:

The only employees authorized to operate vehicle #102 (license plate
TX XXXXXX), vehicle #162 (license plate TX XXXXXX), and vehicle #262
(license plate TX XXXXXX) are those employees who possess the following
driver's licenses:

    TXDL XXXXXX
    TXDL XXXXXX
    VADL XXXXXX
    FLDL XXXXXX

Further, usage of city credit cards will be strictly limited to the
following departmental cards, until further notice:

    MC-3129
    AMEX-8900
    DISC-5389
```

```
26      VISA-1302
27
28  Thank you,
29     Mgmt
30
31  * Policy revision date 2/1/13 (originally passed 7/13/92).
```

Script Execution

Your program should be invoked through a single bash file (see below) with the filename(s) containing the sensitive data as argument(s).

Example: `$ assign2.bash redactme.txt`

Assignment Data

A sample input file can be found in:
`/usr/local/courses/ssilvestro/cs3423/Fall19/assign2`.

When using this data, remember that your will be made to overwrite the files. Be sure to make a backup of the files and restore them every time you run the script.

Script Files

Your program should consist of *exactly* two files:

- `assign2.bash` - the main file which is initially invoked
- *Exactly* one `.sed` file which is used for a sed invocation run in `assign2.bash`.

Verifying Your Program

Your program must work for *arbitrary* files by applying the rules above. You can test your program with the input provided in `redactme.txt` and compare the output with `redacted.txt` using `diff` (check the man-pages on how to use it). You should create your own test cases to test for the recursion feature.

Submission

Turn your assignment in via Blackboard. Your zip file, named `abc123.zip` (with your abc123) should contain only your bash and sed files.