SCRIPTING AND COMPUTER ENVIRONMENTS (CSE505)

BASH SCRIPTING
ASSIGNMENT 1

ASSIGNMENT DEADLINE: 14th September, 2018 11:55 PM (Strict Deadline)

SUBMISSION FORMAT:

1. Create a folder with your roll number `<rollno>_assignment1`

2. Under the folder create `<section no>_<question no>_<subpart>.sh` for each question

3. Compress the file as `<rollno>_assignment1.tar.gz` and then upload it to your moodle account.

HOW TO GET ZERO IN THE ASSIGNMENT?

1. **DEVIATING FROM THE UPLOAD FORMAT** will get you a straight zero.

2. **ANY KIND OF PLAGIARISM** will also lead to a straight zero.

WHERE TO GET HELP?

1. Post your doubts on the Assignment Thread on Moodle.
2. Approach the TAs or visit them in the TA Hours.

---

1. Grep Questions

Question 1.

For the following questions, use the `/usr/share/dict/words` dictionary file.

1. Find all words that have the string India or Africa in them.

2. Find all words that begin with a vowel. How many are there?

---

3. Now, find the words that begin with a vowel and end with a vowel.

4. Find all words that have two or more consecutive a in them.

5. Find all words that have any punctuation marks (such as apostrophe) in them.

6. Find words that contain an uppercase letter.

    (a) at the beginning.

    (b) at the end.

    (c) anywhere.

7. Find all words that are twenty characters or more in length.

8. Find all five-character or ten-character words that do not contain any uppercase letter.

9. Find all words that begin and end with the same letter.(not necessarily the same vowel).

Question 2. Using grep perform following

1. Print count of directories present in current folder.

2. Write Regular expression to match

    (a) xxx.xxx.xxx.xxx where x will be numerals

    (b) same as above but xxx should be less than 256

    (c) valid IP address

3. Write Regular expression to validate mac address for following formats

    (a) Six groups of two hexadecimal digits separated by hyphens (-), like 01-23-45-67-89-ab.

    (b) Six groups of two hexadecimal digits separated by colons (:), like 01:23:45:67:89:ab

(a & b must be validated in a single regular expression)

(c) Verify your regex by extracting mac address of your system using `ifconfig` command.

2. SED Questions

Question 1

Redo the above grep *question-1* and *question-2* using sed command only.

Question 2

You are given the following address book details (in CSV format) in the order:

given: **name**, **surname**, **address**, **city**, **DoB**, **e-mail** (Input File : **address-book.csv**)

Alice, George, 83 First St, Howard, 12/05/1980, awatson@example.com

Bob, Charlie, 40 West Ave, Anycity, 02/11/1995, bcharlie@example.com

Steve, Park, 1981 Songdo, Smartcity, 24/1/1971, spark@exmaple.com

Mary, Sam, 40 West Ave, Anycity, 30/01/1990, msam@example.com

Kumar, Anil, Gachibowli 500032, Hyd, 11/12/1989,kanil@example.com

Write sed commands for the following:

1. Who lives in a city named Anycity?

2. Remove all lines that begin with a vowel(name column).

3. How many people are born in the 80s?

4. Replace all numeric values by ? and all punctuation marks by *.

5. Reverse the order of the Given name and Surname. (e.g. ”Mary, Sam” ”Sam, Mary”).

6. Swap the first and last characters of a line. (e.g. ”Alice ... example.com” ”mlice...example.coA”).

Question 3 - Indent given code

The Minions were given tasks of writing codes but since they never took APS, they dont know about the best indentation practices and Mr. Gru is unable to read the code. Now, your task is to help Mr. Gru to indent programs to some extent. Mr. Gru wants every line to be started with 4 spaces except the header files lines and the lines beginning with { or }. Can you write a sed command to achieve this?

Example Program :

INPUT :

```
int func()
{
int a = 1,b=2;
int c = a+b;
return c;
}
int main()
{
int x,y;
return 0;
}
```

OUTPUT:

```
 int func()
{
   int a = 1,b=2;
   int c = a+b;
   return c;
}
   int main()
{
   int x,y;
   return 0;
}
```

3. AWK Questions

Question 1 - Uniq Awk!

If you have tried the **uniq** command, you'll realise it requires the input to be sorted (can be done using **sort** command). Write an awk command/program that performs the same task as uniq does using awk without needing to sort the file.

(HINT : Try using the uniq command with files/input that have same multiple lines and see how it works)

Question 2 - Grade Reports

You are given the marks of students taking a course named ABC (Input File **marks.txt**).

Name	Gender	Mid1 (25%)	Mid2 (25%)	Endsem (50%)
AA	M	20	15	35
BB	F	22	17	44
CC	F	19	14	25
DD	M	15	20	42
EE	F	18	22	30
FF	M	0	20	45

Now, write an awk command to achieve the following:

1. Display just the names, genders and End Sem marks of:

(a) All students (b) The First 3 students only (c) The Last Student

2. Store the male students in males.txt and the females in females.txt.

3. Compute the total mark for each student and display it along with their names.

4. Who is the topper? Which students scored above the class average?

5. Generate a grade report for each student based on the following marking scheme.

A = [95,100], A- = [90,95), B = [85,90), B- = [80,85)

C = [75,80), C- = [70,75), D = [60,70), F < 60

Your report must contain a header and a footer/end sections as well. The header be entitled, "***** Grade Report for the ABC course *****". At the end of the report, display the total number of students, the highest, lowest & average marks as well as the message, "***** End of Grade Report *****"

Question 3 - Valid JSON

You are given a file named **imdb-top-250.txt**. This file is space separated.

Your task is to convert the above file into JSON format as below:

```
[
    {
    "ID" : "1" ,
    "Name" : "The Shawshank Redemption " ,
    " Year " : " 1994 " ,
    " Rating " : " 9 . 2 "
    } ,
    {
    "ID" : "2" ,
    "Name" : "The Godfather " ,
    " Year " : " 1972 " ,
    " Rating " : " 9 . 2 "
    } ,
    {
    "ID" : "3" ,
    "Name" : "The Godfather : Part I I " ,
    " Year " : " 1974 " ,
    " Rating " : " 9 . 2 "
    } ,
.
.
```

```
.
    {
    "ID" : " 250 " ,
    "Name" : "PK" ,
    " Year " : " 2014 " ,
    " Rating " : " 8 . 0 "
    }
]
```

You are expected to redirect the output to a file named **imdb-top-250.json**. Finally test if it's a valid json by validating it on https://jsonformatter.curiousconcept.com/.

4. BASH Scripting Questions

Question 1. Any Type File Extractor

Write a bash script that takes as input a compressed file as an argument and then extracts the file according to its format. The file can be of any compressed format like .tar,.tar.gz,.tar.bz2,.bz2,.zip etc.

Input Format : bash 4_1.sh <Input_File>

Output : Extract the file according to the compression used.

Perform efficient error handling like checking whether it's a compressed/archived file and a file or not and display proper error messages.

Question 2. Validate Password

Write a script to check and validate passwords. The objective is to flag "weak" for easily guessed password candidates.

A trial password will be input to the script as a command-line parameter. (can use suppressed input type to hide the password) To be considered acceptable, a password must meet the following minimum qualifications:

- Minimum length of 8 characters
- Must contain at least one numeric character
- Must contain at least one of the following non-alphabetic characters: @, #, $, %, &, *, +, -, =

Optional:

- Do a dictionary check on every sequence of at least four consecutive alphabetic characters in the password under test. This will eliminate passwords containing embedded "words" found in a standard dictionary.

- Enable the script to check all the passwords on your system. These do not reside in /etc/passwd.

Question 3. Getting VPN!

After releasing the Assignment, Kaushik (TA) went to his native place (since his home is close by) to spend the free weekend with family. But he found out there have been doubts popping up on the Moodle Thread. To answer this queries, he has to access moodle a lot of times. For that he has to configure the VPN, start the VPN and then login to Moodle. That's when Kaushik realises that he has to repeat this tedious process again and again whenever he goes home. So he thought to get help for this work from the Students of Scripting.

You have to write a bash script (4_3_start.sh) which can download the vpn certificates, extract them to the relevant directories and then start the VPN. There should be another bash file (4_3_stop.sh) for stopping the VPN. Can you guys help Kaushik to write that script?

Requirements:

The bash script (4_3_start.sh) should download the certificates from https://vpn.iiit.ac.in/ (You are not allowed to store them locally). It should then place it in appropriate directory and start the VPN. Once the VPN is started, it should print the message "VPN Started" on the terminal. Later when (4_3_stop.sh) is executed, the VPN should be stopped.

Question 4. Directory Cleaner

Since you download lots of things from internet (like video lectures, notes :P), the Downloads directory gets populated with all sorts of files i.e. pdf, mp4,mp3. To keep things more organized you can periodically move these files from Downloads to some other folder. For e.g. all pdf files would be moved to a folder named pdf. Now write a bash script that can accomplish the same.

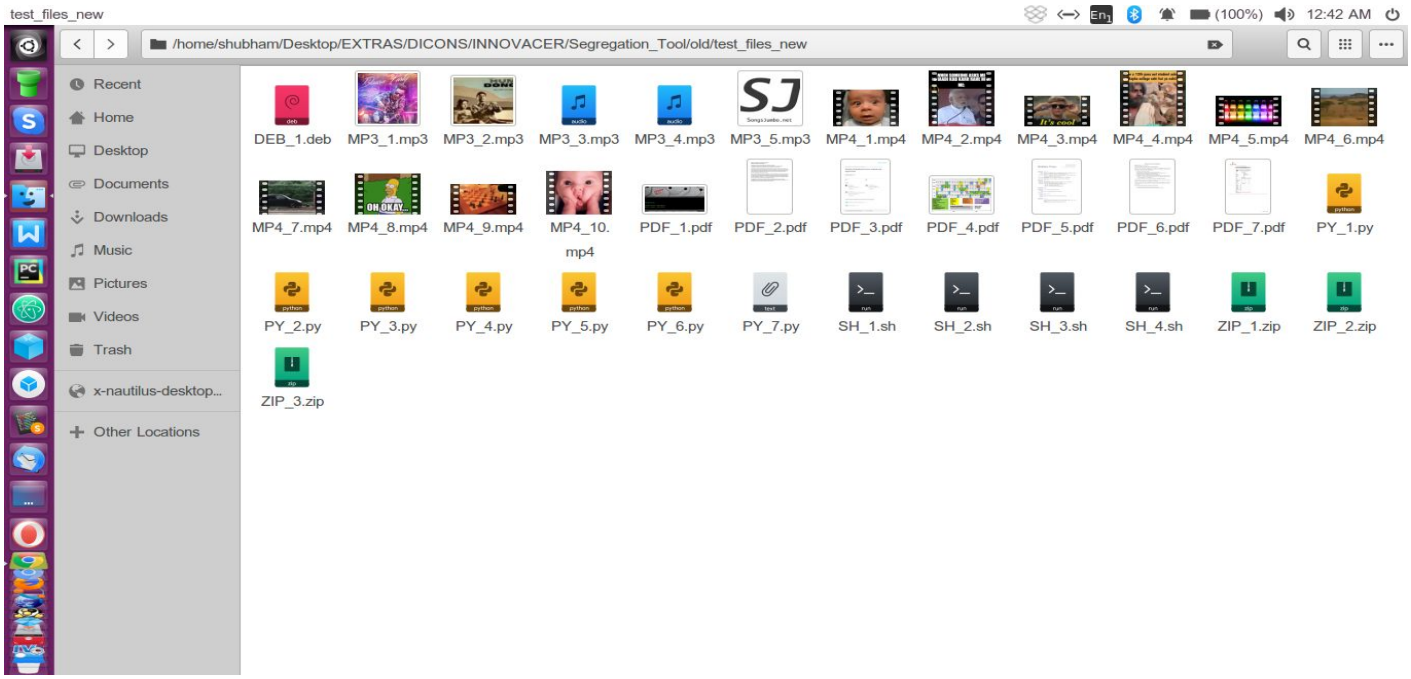
Input : bash 4_2.sh <Source_Directory> <Type_of_Files>

<Source_Directory> => The directory where you have to perform the cleaning
<Type_of_Files> => Extensions of the files to be cleared. Have an **all** option to clean the whole directory.

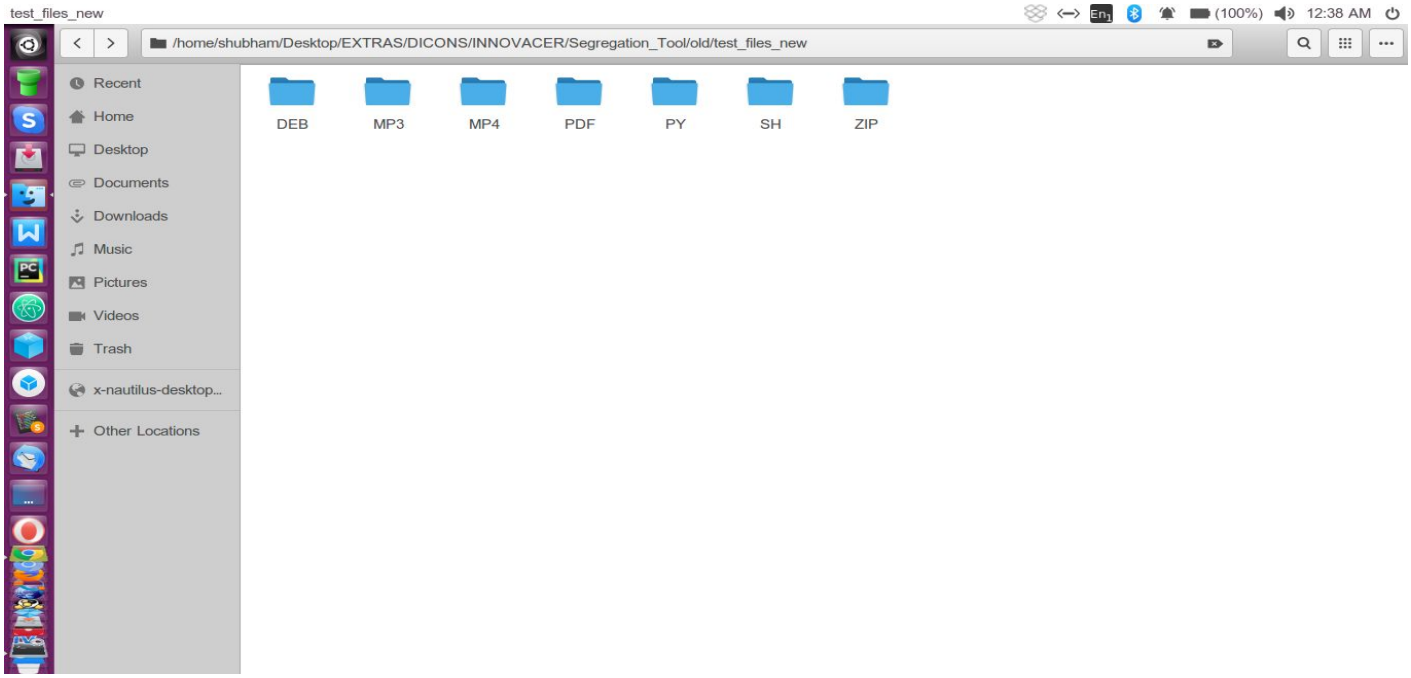
Output : All the files organized into folders with the file extension as the folder names and if there are folders, already present at the location, keep them as it is.

Constraints : Use only bash to perform this. Here is an example

Before:



After:



Expectations :

(1) Efficient Error Handling. (Various cases like if the source directory is invalid etc).

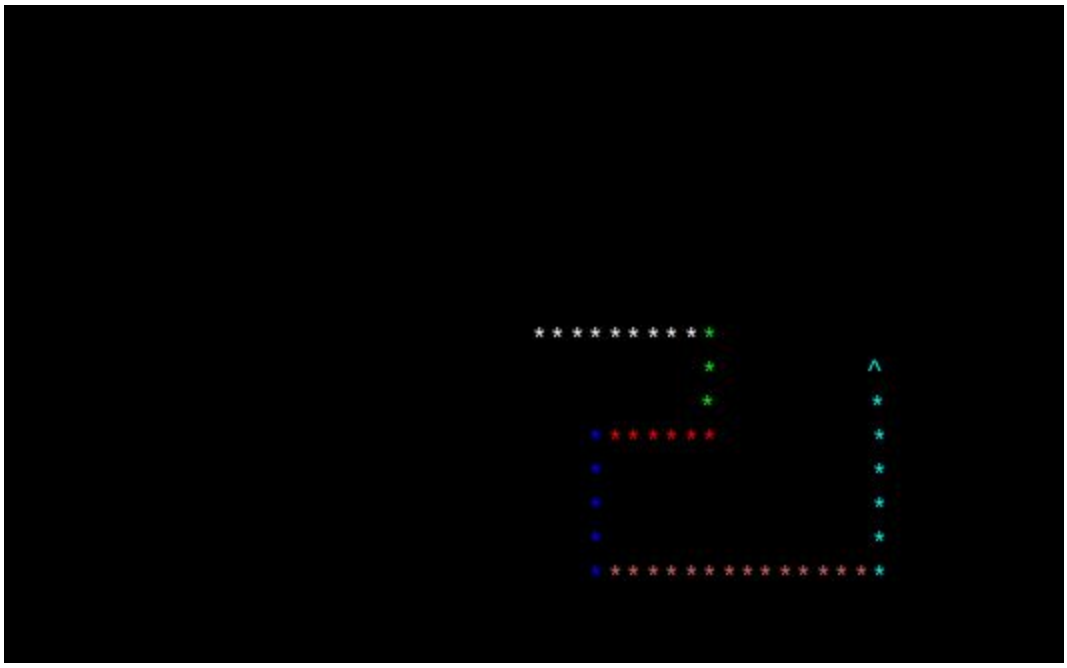
(2) Not modifying the existing folders in the directory.

Bonus Part : If you find this question easy, try making this operation recursive.

Question 5. Snake Game

We are the 90's kids so each one of us must have played the SNAKE GAME once in our life. What about making one for yourself. Relax you are not required to make the whole snake game but just a part of it.

1) You are required to move the snake in any of the four directions using the arrow keys.
2) The length of the snake can increase periodically. The rate at which the length of the snake increases can be set by you just make sure it looks fine.
3) When the snake reaches the end of any of the four sides of the terminal then It should Restart from middle.
4) The colour of the snake should change for every change in direction randomly.



SOME HINTS :

1. For coloring the text inside the terminal, find out about **tput** command.
2. Find out how to detect arrow keys in bash to make the snake move using arrow keys.

SOME WARNINGS :

1. It is a tough question, so we will also be judging your creativity with it.
2. Don't copy paste code. Plagiarism detected in this question will lead to zero in the whole assignment.

Happy Scripting!!