

EASY (10 Points each)

1. Write a bash script that echoes itself to stdout, but backwards.
2. Write a bash script that backs itself up, that is, copies itself to a file named backup.sh.
3. Write a script to *lowercase* all file/directory names in a directory. Input will be a single directory. If a lowercase file/directory of the same name already exists in that directory, the program should warn the user and NOT overwrite the existing file/directory.
4. Solve a quadratic equation of the form  $Ax^2 + Bx + C = 0$ . Have a script take as arguments the coefficients, A, B, and C, and return the solutions to five decimal places.

INTERMEDIATE : ( 15 Points each)

5. Write a recursive script that recursively lists all files i.e is equivalent to 'ls -R'. Arguments will work like those provided to the ls command. (Hint: Use export and readlink to store absolute path of the script)
6. Write a script that produces a visualization tree of a directory. You must take only 1 argument as input and it must be a directory.
7. Assume that you are an administrator of a lab machine. The labs are open on all days except Sunday.

Write a shell script that logs all accesses to the files in /etc during the course of a single day (Take it to be the previous day and NOT 24 hrs from now).

This information should include the filename and access time. Write this data as tabular (tab-separated) formatted records in a logfile called "AccessLog".

(Hint: Use sudo to get over 'permission denied' messages)

Extra Credit : (5 Points ~ Only if both are answered )

- 1) Find a way to run this everyday at 12:01 AM or 0001 hrs.
- 2) Find a way to run this every monday and generate a log for accesses made on sundays.

8. Implement, as a script, a "safe" delete command, 8.sh. Filenames passed as command-line arguments to this script are not deleted, but instead gzipped if not already compressed (use file to check), then moved to a ~/TRASH directory. Upon invocation, the script checks the ~/TRASH directory for files older than 48 hours and permanently deletes them.

Extra credit (5 Points):

Rewrite this script, which can handle files and directories recursively. This would give it the capability of "safely deleting" an entire directory structure.

9. Write a script that implements C String operations strcmp, strcat, strlen, strtok, strstr, and use it in another script that reads 2 strings and then performs all these operations.

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

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: @, #, \$, %, &, \*, +, -, =

Do a dictionary check (Use /usr/share/dict/words) 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.

The input to your script will be a file containing passwords, one per line. You must generate an output file 'passwordtest.txt' whose lines will say WEAK or STRONG corresponding to the input password line.

**11. (M.Tech CSE Only)**

**A file contains list of roll numbers of students who copied an assignment from each other. Write a shell script which reads the file and outputs groups of all students who copied the assignments together. For example, if file has:**

**201305641 201305581**

**201305641 201305051**

**201305051 201305581**

**201305051 201305021**

**201305021 201305051**

**201306532 201305111**

**201306532 201205121**

**201305641 201205874**

**201305532 201305182**

**then output would be:**

**201305532 201305182**

**201305641 201305051 201305581 201305021 201205874**

**201306532 201305111 201205121**

**12. (M.Tech VLSI Only)**

**Consider a hypothetical machine :**

**(i)This machine has 5 registers of 64 bit each,viz. AX BX CX DX EX where AX is an accumulator while the rest are general purpose. All of them must store data in binary.**

**(ii)Given an assembly code for this machine, with a set of instructions to implement, write a script that evaluates each of these instructions.**

### **13. (M.Tech CSIS Only)**

**Perform encryption and decryption.**

**Given a file with text in it (having a-z A-Z 0-9, space and dot(.)), encrypt the file using the scheme given below (next page), write encrypted text in a separate file. Encrypted File name should be 13\_Encrypted.txt**

**Read the encrypted data from this file and decrypt it using reverse scheme and write the decrypted data into another file. File name should be 13\_Decrypted.txt .Make sure your original file data and decrypted file data match.**

**Make separate functions to encrypt and decrypt the data and write them in a single script file 13.sh**

### **14) (M.Tech BI Only)**

**You are given two files "14\_ref.txt" and "14\_sample.txt" which contain a part of Illumina mRNA chip expression data for the BRCA1 gene. This gene is a tumor suppressor gene and is involved in resistance to Breast Cancer in Humans.**

**Write a shell script for the following:**

**a. Distance calculation b/w corresponding cells in the expression data files. e.g. : for first cell (ILMN\_1725881), the values will be 196.6413 and 187.7823. Write the distance values for each cell to a file : "output.txt". Use minkowski distance calculating algorithm.**

**b. Cluster the cells into 4 clusters using distance values calculated in step a. Perform unweighted hierarchical clustering. Use any threshold of your choice. Write the cluster number and the values of that cluster to the file 'output.txt' (use of any simple approach for clustering will be fine).**

**c. Calculate the median distance between each of the 4 clusters.**

**d. Find the total distance b/w the genes. Assuming that each cell had 30 consecutive bp's of BRCA1 gene (from start) and using all the results, what all inferences can you derive about :-**

**- conserved regions in the gene, - positions of mutation in the gene , - extent of divergence**