1. (10 points)
vim helps you to edit files on your system. If you supply a filename, vim tries to open the file when it starts. For example;

      $ vim /home/ilker/scripts/aSampleScript.sh

Assume you have a script you want to edit, named *myScript.sh*, but you don't remember where you have moved it. The script runs when you type its name, so it must be in your PATH. Write a <u>single line command</u> that loads the script file to **vim**.

2.
a. (10 points)
Write a script which gives the below output. You should use nested loop structures.

 sample run: $./sampleScript.sh
        #
        # #
        # # #

b. (10 points)
Modify your answer to question 2.a in such a way that your script takes an argument as the string to be displayed as in the same format of the output of question 2.a.

 sample run: $./modifiedSampleScript.sh CE350
        CE350
        CE350 CE350
        CE350 CE350 CE350

3. (10 points)
Name two UNIX/Linux system environment variables and what they are used for.

4. (10 points)

Give file *mySecretData.dat*

       a. read and write permissions only for its owner.

       b. read, write, execute permissions for its owner, read and write permissions for the group and only executable permission for others.

NOTE: Use decimal numbering system for permissions.

5.

a. (10 points)

One of your non-computer-savvy friends has just decided to use UNIX. Help him/her by creating a **REN** command. On DOS, REN command renames a file. So, the command:

       REN f1.txt f2.txt

renames f1.txt to f2.txt.

b. (10 points)

How can you make above REN command permanently available to use?

---

6. (15 points)

If a user tries the following commands in the working directory, what might be the aim of the user?

       a. $ ls abc[123]

       b. $ ls -l [ab]*

       c. $ grep -f '$HOME' myFile

       d. $ grep -r 'ilker' /home

       e. $ who | grep ilker > outFile

---

7. (15 points)

Write a script called *checking* that will do the following:

       i. Take a command-line argument, a user's login name.

       ii. Test to see if a command-line argument was provided.

       iii. Check to see if the user is in the /etc/passwd file. If so, print "Found <user> in the /etc/passwd file"; otherwise, print "No such user on our system".