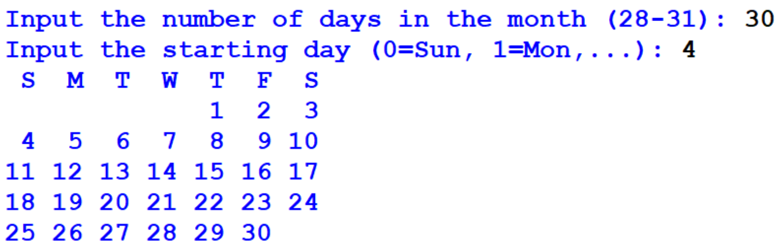
Lab Exercise - 10

Q.1. Using for loop, write and run a Python program for this algorithm. Here is an algorithm to print out n! (n factorial) from 0! to 10! :

```
1. Set f = 1
2. Set n = 0
3. Repeat the following 10 times:
a. Output n, "! = ", f
b. Add 1 to n
c. Multiply f by n
```

Q.2. Modify the program above using a while loop so it prints out all of the factorial values that are less than 2 billion. (You should be able to do this without looking at the output of the previous exercise.)

Q.3. Write a program that asks the user how many days are in a particular month, and what day of the week the month begins on (0 for Monday, 1 for Tuesday, etc), and then prints a calendar for that month. For example, here is the output for a 30-day month that begins on day 4 (Thursday):



```
Input the number of days in the month (28-31): 30
Input the starting day (0=Sun, 1=Mon,...): 4
 S  M  T  W  T  F  S
             1  2  3
 4  5  6  7  8  9 10
11 12 13 14 15 16 17
18 19 20 21 22 23 24
25 26 27 28 29 30
```

Q. 4. Define a procedure histogram() that takes a list of integers and prints a histogram to the screen. For example, histogram([4, 9, 7]) should print the following:

```
****
*********
*******
```

Q. 5. Write a version of a palindrome recognizer that also accepts phrase palindromes such as "Go hang a salami I'm a lasagna hog.", "Was it a rat I saw?", "Step on no pets", "Sit on a potato pan, Otis", "Lisa Bonet ate no basil", "Satan, oscillate my metallic sonatas", "I roamed under it as a tired nude Maori", "Rise to vote sir", or the exclamation "Dammit, I'm mad!". Note that punctuation, capitalization, and spacing are usually ignored.

Q. 6. A pangram is a sentence that contains all the letters of the English alphabet at least once, for example: The quick brown fox jumps over the lazy dog. Your task here is to write a function to check a sentence to see if it is a pangram or not.

Q. 7. In cryptography, a Caesar cipher is a very simple encryption techniques in which each letter in the plain text is replaced by a letter some fixed number of positions down the alphabet. For example, with a shift of 3, A would be replaced by D, B would become E, and so on. The method is named after Julius Caesar, who used it to communicate with his generals. ROT-13 ("rotate by 13 places") is a widely used example of a Caesar cipher where the shift is 13. In Python, the key for ROT-13 may be represented by means of the following dictionary:

key = {'a':'n', 'b':'o', 'c':'p', 'd':'q', 'e':'r', 'f':'s', 'g':'t', 'h':'u', 'i':'v', 'j':'w', 'k':'x', 'l':'y', 'm':'z', 'n':'a', 'o':'b', 'p':'c', 'q':'d', 'r':'e', 's':'f', 't':'g', 'u':'h', 'v':'i', 'w':'j', 'x':'k', 'y':'l', 'z':'m', 'A':'N', 'B':'O', 'C':'P', 'D':'Q', 'E':'R', 'F':'S', 'G':'T', 'H':'U', 'I':'V', 'J':'W', 'K':'X', 'L':'Y', 'M':'Z', 'N':'A', 'O':'B', 'P':'C', 'Q':'D', 'R':'E', 'S':'F', 'T':'G', 'U':'H', 'V':'I', 'W':'J', 'X':'K', 'Y':'L', 'Z':'M'}

Your task in this exercise is to implement an encoder/decoder of ROT-13. Once you're done, you will be able to read the following secret message:

Pnrfne pvcure? V zhpu cersre Pnrfne fnynq!

Note that since English has 26 characters, your ROT-13 program will be able to both encode and decode texts written in English.

Q. 8. Given a dictionary of students and their favourite colours:

people={'Arham':'Blue','Lisa':'Yellow',''Vinod:'Purple','Jenny':'Pink'}

1. Find out how many students are in the list
2. Change Lisa’s favourite colour
3. Remove 'Jenny' and her favourite colour
4. Sort and print students and their favourite colours alphabetically by name

Q 9: Write a function translate() that will translate a text into "rövarspråket" (Swedish for "robber's language"). That is, double every consonant and place an occurrence of "o" in between. For example, translate("this is fun") should return the string "tothohisos isos fofunon".

Q. 10. Write a program that contains a function that has one parameter, n, representing an integer greater than 0. The function should return n! (n factorial). Then write a main function that calls this function with the values 1 through 20, one at a time, printing the returned results. This is what your output should look like:

```
1 1
2 2
3 6
4 24
5 120
6 720
7 5040
8 40320
9 362880
10 3628800
```

Q. 11. We can define sum from 1 to x (i.e. 1 + 2 + ... + x) recursively as follows for integer x ≥ 1:

1, if x = 1

x + sum from 1 to x-1 if x > 1

Complete the following Python program to compute the sum 1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 + 10 recursively:

```
def main():
# compute and print 1 + 2 + ... + 10
```

```
print sum(10)
def sum(x):
# you complete this function recursively main()
```

Q. 12. Define a function overlapping() that takes two lists and returns True if they have at least one member in common, False otherwise.

Q. 13. Write a function find_longest_word() that takes a list of words and returns the length of the longest one.

Q. 14. Write a function filter_long_words() that takes a list of words and an integer n and returns the list of words that are longer than n

Q. 15. Define a simple "spelling correction" function correct() that takes a string and sees to it that

1) two or more occurrences of the space character is compressed into one, and

2) inserts an extra space after a period if the period is directly followed by a letter.

e.g. correct("This is very funny and cool.Indeed!") should return "This is very funny and cool. Indeed!"

Q. 16. In English, present participle is formed by adding suffix -ing to infinite form: go -> going. A simple set of heuristic rules can be given as follows:

- If the verb ends in e, drop the e and add ing (if not exception: be, see, flee, knee, etc.)
- If the verb ends in ie, change ie to y and add ing
- For words consisting of consonant-vowel-consonant, double the final letter before adding ing
- By default just add ing

Your task in this exercise is to define a function make_ing_form() which given a verb in infinitive form returns its present participle form. Test your function with words such as lie, see, move and hug. However, you must not expect such simple rules to work for all cases.

Q17: Write a Python program to remove leading zeros from an IP address.

For example:

IP Address **216.08.094.196** should be represented as **216.8.94.196**