**Gebze Technical University**
**INF100 Introduction to Computer Systems**
**Homework**
**Fall 2022**

---

- Submission Deadline : **13<sup>th</sup> January 17:00**. No late submissions will be accepted.

- Hand-in Policy: via Teams.

- For each question a Python code together with a one-paragraph report is required.

- All the codes must be included in a single Colab file with the extension **.ipynb** and submitted via Teams.

---

1. Write a function **find_grade_frequency()** which takes a dictionary containing elements in the form of student name : the letter grade as input and returns a dictionary including elements in the form of the letter grade : frequency of the letter grade. Note that the keys in this dictionary must be alphabetically ordered as shown in expected output. Make a dictionary like the one given in sample input and pass it to your function in your main code and print the result.

   **Sample input**:

```
Grades= {'Selda'    : 'cb',
         'Ali'      : 'aa',
         'Muhammet' : 'ba',
         'Zehra'    : 'ba',
         'Pelin'    : 'cc',
         'Ahmet'    : 'ba',
         'Can'      : 'aa',
         'Emre'     : 'cb',
         'Yakup'    : 'bb',
         'Selin'    : 'cb',
         'Alper'    : 'aa'
}
```

   **Expected output**:

```
{'aa': 3, 'ba': 3, 'bb': 1, 'cb': 3, 'cc': 1}
```

2. Design a recursive function **print_digits()** which takes a positive decimal number as an input and prints its digits from the most important one to the least important one. Call this function in your main code with an arbitrary number taken as input from the user and print the result.

   **Sample input and output**:

```
print_digits(14326)

1
4
3
2
6
```
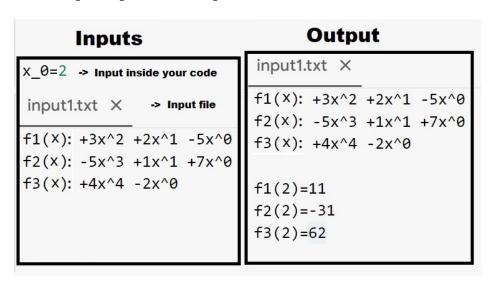
3. Write a Python code which takes two square matrices $A$ and $B$ as inputs in the form of lists of lists and displays $A \times B$. Your code should have two functions: One **matrix_multiplication()** function for calculating the multiplication of the input matrices and one **matrix_display()** function which displays the result in the form similar to a matrix. For the display function, you can use <u>the one which we had in one of our lab sessions</u> or you can write your own display function.

   **Sample input and output**:

```
Inputs:                                    result:
A=[[1,4,5],[7,6,4],[3,2,3]]                | 21 26 42 |
                                           | 32 54 87 |
B=[[2,4,5],[1,3,8],[3,2,1]]                | 17 24 34 |
```

4. Write a code which reads a text file containing polynomials $f_1, f_2, ..., f_k$ and takes an input value $x_0$ from a user and calculates the values $f_1(x_0), f_2(x_0), ..., f_k(x_0)$ and writes these values to the same text file according to the format shown in the following:

   **Sample input and output**:

```
Inputs                              Output

x_0=2  -> Input inside your code    input1.txt  ✕

input1.txt  ✕        -> Input file  f1(x): +3x^2 +2x^1 -5x^0
                                    f2(x): -5x^3 +1x^1 +7x^0
f1(x): +3x^2 +2x^1 -5x^0            f3(x): +4x^4 -2x^0
f2(x): -5x^3 +1x^1 +7x^0
f3(x): +4x^4 -2x^0                  f1(2)=11
                                    f2(2)=-31
                                    f3(2)=62
```

   **Note 1**: There is at least one space between the terms of each polynomial $f_i(x)$ in the input text file. Use these spaces for splitting purposes.
   **Note 2**: You are allowed to use Python's **pow()** function when calculating $f_i(x_0)$.