CSE101: Introduction to Programming
Lab 9

This lab would require you to perform some data processing to rearrange data based on an ordering scheme.

Create a module, **StudentRecord.py**.
Create a **Student** class in it that initializes the Student object with a student's
Rollnumber, FirstName, LastName, Program, CGPA
Where

- Rollnumber is a unique 7 letter code of string type with first 4 letters representing the Batch year. Eg. 2016022
- FirstName, LastName are of string type
- Program is a string representing the degree program of the student eg. 'B.Tech-CSE'
- CGPA is of double type in the range 0-10

Assume that none of these entities contain any spaces.
In this class define a method, **comes_before(self,other)**, which returns True if current Student object comes before the other Student object(in an ordering scheme as described below), else it returns False.
A Student, S1, is ordered before another Student,S2, based on following priority rule:

- S1 took the admission before S2 i.e Batch year of S1 comes before that of S2
    - Incase, S1 and S2 belong to same batch, then order by the Program in a lexicographic order. Eg. 'B.Tech-CSAM' comes before 'B.Tech-CSE'
        - Further, if S1 and S2 belong to same Programs(and batch) as well then order by their CGPA. Student with higher CGPA should come before the other student. For simplicity, Assume CGPA would be unique for everyone.

You have been provided with a text file, **studentdata.txt**, containing record of students with all the details available row wise in this format for every student:

Rollnumber FirstName LastName Program CGPA

However, the rows are randomly ordered. Your task is to arrange the content of the file according to the order scheme specified previously. Here are your tasks to achieve this:

Create another module,**ArrangeRecords.py**, that imports StudentRecord and define the following functions in it:

- **readrecords(*filename*)**: reads the data from the text file and forms a Student object for each record. The function returns a list of Student objects.

- **order_records(*studentlist*)**: Takes the list returned by readrecords function and orders the Student objects according to the Ordering scheme defined previously. You may have to use the comes_before method defined in Student class to perform this ordering. order_records returns a list of ordered Student objects.

- **display_ordered_data(*orderedlist*)**: The function takes ordered list returned by order_records function and displays/prints the Student data arranged in order. The data for each Student should be in newline and should include the following details, each separated by a single space.
  Rollnumber FirstName LastName Program CGPA

Submission Instructions:
You must mention your **name and roll number** as comments in both the .py files.
Submit StudentRecord.py and Arrange Records.py in a zipped file