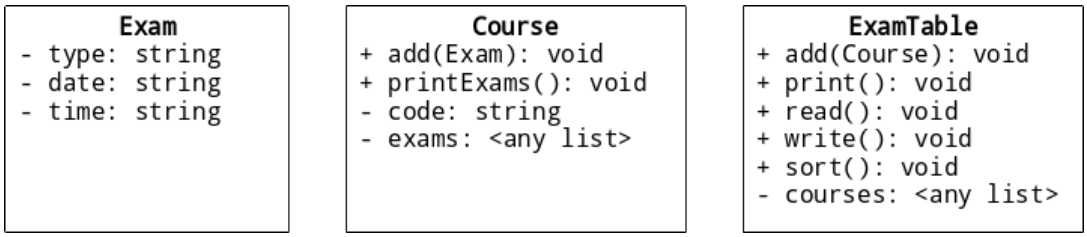
CENG218 Homework 1

In this homework, you are expected to develop an exam schedule table system using C++ programming language. With this system, users must be able to:

1. Add new courses to the table,
2. Delete existing courses from the table,
3. Add exams to existing courses in the table,
4. Modify existing exams in the table,
5. Delete existing exams from the table,
6. Display the table,
7. Save current table to a file on exit,
8. Load previously saved table from a file on startup.

You are expected to implement three classes for this system:



- **Exam**:
  - **type**, **date**, **time**: simple strings to store appropriate values (e.g. "Midterm", "27.04.2022", "17:30", etc).
- **Course**:
  - **add()**: Adds a new Exam object to the *exams* list.
  - **printExams()**: Iterates over the *exams* list and prints them one by one on screen.
  - **code**: a string that stores course code (e.g. CENG218).
  - **exams**: a list of **Exam** objects. You are free to implement any time of list for this variable (static array, dynamic array, vector, etc).
- **ExamTable**:
  - **print()**: Prints a sorted list of exams by iterating over the *courses* list and calling the *printExams()* function of each course.
  - **read()**: Reads a previously saved exam list from file into memory.
  - **write()**: Writes the exam list in memory into a file.
  - **sort()**: Sorts the *courses* list by course code.
  - **courses**: a list of **Course** objects. You are free to implement any time of list for this variable (static array, dynamic array, vector, etc).

Above are some of the suggested member variables and functions for Exam, Course and ExamTable classes. In addition to the ones listed above, each class must also have constructors and getters/setters as appropriate. You are expected to define additional member variables and implement additional member functions in these classes as necessary.

You are expected to deliver a well implemented C++ program that must be able to be compiled and run without producing any errors or warnings. Explain your functions and code segments using comment lines. Your program is expected to run similar (but not strictly the same) to the sample run presented below. Study it carefully. User inputs are underlined.

Sample Run

```
1. Display table
2. Add new course
3. Delete existing course
4. Add exam to existing course
5. Delete existing exam
6. Exit
Your choice: 1
Course  Type    Date       Time
(none)

... Your choice: 2
Enter course code: CENG218
Course added.

... Your choice: 2
Enter course code: CENG114

... Your choice: 4
Available courses:
1. CENG114
2. CENG218
Which course? 2
Enter exam type, date and time: Midterm 28.03.2022 17:30

... Your choice: 4
Available courses:
1. CENG114
2. CENG218
Which course? 1
Enter exam type, date and time: Final 07.06.2022 12:30

Your choice: 1
Course  Type    Date       Time
CENG114 Final   07.06.2022 12:30
CENG218 Midterm 28.03.2022 17:30

... Your choice: 6
Bye!
```

Contents of the saved text file (e.g. exams.txt) at this moment can be like:

```
CENG114 Final 07.06.2022 12:30
CENG218 Midterm 28.03.2022 17:30
```