EECS 2031 Program Exam 2 -- Due: April 5th 2021 23:59 pm

This programming exam is going to assess your ability for the following skills:

- Ability to use string library functions and formatted IO library functions.
- Ability to manipulate pointers, structures, array of structures and array of pointers to structures
- Ability to do dynamic memory allocation when needed
- Basic coding style

Coding environment:
Similar to the PE1, you can work by connecting to the prism lab, or, work locally. Make sure the final deliverable will compile in the lab environment.

In this assignment, you are to develop a student enrolment management system for our EECS department. The purpose of this assignment is to help you better understand some of the advanced concepts in C that we covered recently. These include array of pointers, structures, pointer to structures, dynamic memory allocation, formatted IO, etc.

Download file `PE2.c` to start off. Study the provided codes.

The enrollment system maintains a database, which maintains a collection of EECS student records. Each student record is naturally implemented as a structure, and contains the following information fields:

- `name` of type "string" `(char [])`, which represents the student's name
- `age` which represents the student's age
- `course-1` of type "string" which represents a course that the student is enrolled in
- `course-2` of type "string" which represents the other course that the student is enrolled in (Assume each student enroll in exact two courses)
- `status` of type "string", which represents the systems' feedback about the course enrollment, in terms of whether there is a time conflict of the two courses.

Programmatically, the database maintains an **array of pointers to record structs,** as shown in Figure 1.

The database also maintains a collection (array) of courses structs, as shown in Figure 2. Each course struct contains course code, course title, lecture date, start time, ending time and location. This collection of course is loaded from a disk file `course.txt`. To make the code simpler, this array of course is declared as a global variable.

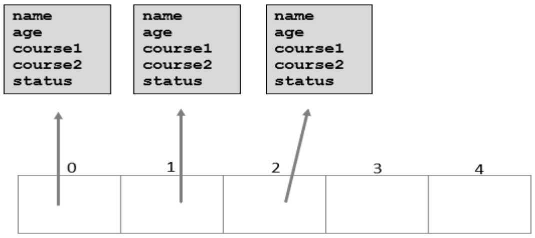

Figure 1 Main database: Array of pointers to record (structure)

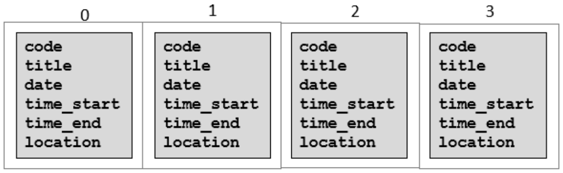

Figure 2 Array of course structures

The program will provide the following basic functionalities:

- Displaying information of all or a particular course offered in the department.
- Entering a new student record into the current database
    - allow the students to enroll in two courses, and generate feedbacks about whether there is a time conflict about the two courses.
- Displaying all student records in the current database
- Removing an existing student's record in the current database
- Swapping the records in the current database
- Sorting all the records in the current database
- Clearing the current database
- Saving current database to the disk file
- Loading database from the disk file
- Emptying the disk file

Specifically, when the program starts, it first reads the data file `courses.txt` (assumed to be in the same directory), and stores the course information in the file into the array of `course` structures (implemented for you). Note that, the program adds a special sentinel course structure with code "`EECS0000`" into the course array, indicating the end of the array.
The program then keeps on prompting the user with the following menu, until `q` or `Q` is chosen, which terminates the program.
The program should fulfill the following functionalities (some have been implemented for you):

- Keeps on prompting and responding to user inputs. Valid input includes `V/v`, `N/n`, `D/d`, `L/l`, `W/w`, `E/e`, `S/s`, `C/c`, `R/r`, `P/p` and `Q/q`. Displays error messages for other inputs as shown below. (This has been implemented for you.)

```
red 325 % gcc PE2.c
red 326 % a.out
-------------------------------------------------------------------
|    (N)ew record          (R)emove record      Swa(p) records      |
|    (S)ort database       (C)lear database     (D)isplay db        |
|    (L)oad disk           (W)rite disk         (E)mpty disk        |
|    (V)iew courses        (Q)uit               *Case Insenstive*   |
-------------------------------------------------------------------
choose one: x
not a valid input!
```

```
-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insenstive*  |
-----------------------------------------------------------------
choose one: sort
not a valid input!

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one:
```

- When user chooses `V` or `v`, which represents `View Courses`, further prompts for course number or `a`. If the user enters `a`, the program displays all the current available courses. If the user enters a course number, which is assumed to be in the form of either `EECSXXXX` (e.g., `EECS2021`) or just `XXXX` (e.g., `2031`), then the program searches for the course, and display the information of the course if the course exists. If the course number does not exist, display some error message, as illustrated below.

```
-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: v
course code (or 'a')? a

==================================================================================
EECS1001        Research Directions in Computing        W     16:30-17:30   VC 135
EECS1021        OOP from Sensors to Actuators           MWF   10:30-11:30   ACE 009
EECS1022        Programming for Mobile Computing        MW    17:30-19:00   CLH A
EECS2001        Intro to the Theory of Computation      MW    14:30-16:00   CLH M
EECS2011        Fundamentals of Data Structures         TR    13:00-14:30   LSB 106
EECS2021        Computer Organization                   MW    17:30-19:00   LAS B
EECS2031        Software Tools                          T     16:30-18:30   SLH E
EECS2200        Electrical Circuits                     W     11:30-13:30   CB 120
EECS3101        Design and Analysis of Algorithms       MW    11:30-13:00   SLH C
EECS3213        Communication Networks                  MW    17:30-19:00   BC 215
EECS3214        Computer Network Protocols              TR    10:00-11:30   TEL 1005
EECS3215        Embedded Systems and Design protocol    TR    16:00-17:30   CB 120
EECS3221        Operating System Fundamentals           TR    14:30-16:00   HNE B15
EECS3311        Software Design                         MW    16:00-17:30   R N203
EECS3401        Functional & Logic Programming          MW    16:00-17:30   HNE B15
EECS3421        Introduction to Database Systems        TR    17:30-19:00   HNE B15
EECS4101        Advanced Data Structures                MW    16:00-17:30   CLH M
EECS4111        Automata and Computability              MW    11:30-13:00   CB 122
EECS4215        Mobile Communications Networks          TR    17:30-19:00   CB 122
EECS4431        Advanced Topics in 3D                   TR    14:30-16:00   CB 120
EECS4471        Introduction to Virtual Reality         MW    10:00-11:00   CB 122
====================================================================================
-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: v
```

```
course code (or 'a')? 3433
error! course does not exist
-----------------------------------------------------------------
|       (N)ew record        (U)pdate record     Swa(p) records   |
|       (S)ort database     (C)lear database    (D)isplay db     |
|       (L)oad disk         (W)rite disk        (E)mpty disk     |
|       (V)iew courses      (R)emove record     (Q)uit           |
-----------------------------------------------------------------
choose one: v
course code (or 'a')? 2031
EECS2031        Software Tools                            T     16:30-18:30   SLH E

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: v
course code (or 'a')? EECS2021
EECS2021        Computer Organization                     MW    17:30-19:00   LAS B

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one:
```

Implement the function `void displayCourse(struct db_type * pArr[])` to accomplish this.

- When the users enters `n` or `N`, which represents `New record`, the program further prompts the user to enter a new student record into the database. The program prompts and reads in student's name, age, code of course1 and course2, and create a new record with this information. Assume the user enters a course code in the form of either `EECSXXXX` or `XXXX`. If the course does not exist in the collection of courses, prompts the user to enter again until an existing course code is entered. If the course exists, copy the course code, title, date, time and location from the course collections into the `course1` or `course2` member of the new record (as formatted string). It also generates a feedback about possible time conflict of the two courses. Time conflict occurs if on any of the lecture days there are time overlap. The feedback is stored in the `status` member of the new record and is also displayed on screen. The program then inserts the new record into the database (array of pointers to student structures). Assume that the student name in the new record is different from all the names of the existing records in the database (and thus no duplicate student names exist in the database). Also assume that the two courses enrolled for each record are different. Also assume the database is not full.
- When the user enters `d` or `D`, displays the current database of (all) student records.
- When the user enters `c` or `C`, clear the current database (implemented for you).

Sample inputs/outputs involving `n/N` and `d/D` and `c/C` are shown below.

```
-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: d
===============================
========== 0 records ==========
-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: n
name: Judy Sue
age: 22
course-1: 1011
course does not exist, enter again: EECS1020
course does not exist, enter again: 1021
course-2: 1028
course does not exist, enter again: 1029
course does not exist, enter again: EECS1022
SUCCESSFUL! no time conflict

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: d
===============================

name:    Judy Sue
age:     22
course1: EECS1021       OOP from Sensors to Actuators           MWF   10:30-11:30   ACE 009
course2: EECS1022       Programming for Mobile Computing        MW    17:30-19:00   CLH A
remarks: SUCCESSFUL! no time conflict

========== 1 records ==========

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: n
name: John Lee
age: 24
course-1: 2031
course-2: EECS4215
ATTENTION! time conflict

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: d
===============================

name:    Judy Sue
age:     22
course1: EECS1021       OOP from Sensors to Actuators           MWF   10:30-11:30   ACE 009
course2: EECS1022       Programming for Mobile Computing        MW    17:30-19:00   CLH A
remarks: SUCCESSFUL! no time conflict

name:    John Lee
```

```
age:     24
course1: EECS2031       Software Tools                          T     16:30-18:30   SLH E
course2: EECS4215       Mobile Communications Networks          TR    17:30-19:00   CB 122
remarks: ATTENTION! time conflict

========== 2 records ==========

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: C
are you sure to clear db? (y) or (n)? y

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: d
===============================
========== 0 records ==========

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one:
```

Implement function **`void enterNew(struct db_type * pArr[])`** and **`void displayDB(struct db_type * pArr[])`** to accomplish the above. Add any helper functions as needed.

- When user chooses `R` or `r`, which represents `Remove record`, further prompts the user for the name of the student whose record is to be removed. If no record by that name is found in the current database, then the error message "record not found" should be printed out. If the record is found, the record is removed from the database, with message "record [xx] removed successfully". Note that after removal, the relative ordering of the remaining records should remain unchanged, as shown below.

```
-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: d
===============================

name:    Alice Zue
age:     20
course1: EECS1021       OOP from Sensors to Actuators           MWF   10:30-11:30   ACE 009
course2: EECS4471       Introduction to Virtual Reality         MW    10:00-11:00   CB 122
remarks: ATTENTION! time conflict
```

```
name:    Bill Las
age:     21
course1: EECS2001       Intro to the Theory of Computation      MW    14:30-16:00   CLH M
course2: EECS2031       Software Tools                          T     16:30-18:30   SLH E
remarks: SUCCESSFUL! no time conflict

name:    Cindy Sue
age:     33
course1: EECS3101       Design and Analysis of Algorithms       MW    11:30-13:00   SLH C
course2: EECS2031       Software Tools                          T     16:30-18:30   SLH E
remarks: SUCCESSFUL! no time conflict

name:    Dusan Luc
age:     33
course1: EECS2021       Computer Organization                   MW    17:30-19:00   LAS B
course2: EECS2031       Software Tools                          T     16:30-18:30   SLH E
remarks: SUCCESSFUL! no time conflict

========== 4 records ==========

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: r
enter full name to remove: Linh Ngu
record not found

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: r
enter full name to remove: Bill Las
record [Bill Las] removed successfully.

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: d
===============================

name:    Alice Zue
age:     20
course1: EECS1021       OOP from Sensors to Actuators           MWF   10:30-11:30   ACE 009
course2: EECS4471       Introduction to Virtual Reality         MW    10:00-11:00   CB 122
remarks: ATTENTION! time conflict

name:    Cindy Sue
age:     33
course1: EECS3101       Design and Analysis of Algorithms       MW    11:30-13:00   SLH C
course2: EECS2031       Software Tools                          T     16:30-18:30   SLH E
remarks: SUCCESSFUL! no time conflict

name:    Dusan Luc
age:     33
```

```
course1: EECS2021       Computer Organization                   MW    17:30-19:00   LAS B
course2: EECS2031       Software Tools                          T     16:30-18:30   SLH E
remarks: SUCCESSFUL! no time conflict

========== 3 records ==========

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: r
enter full name to remove: Cindy
record not found

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: R
enter full name to remove: Cindy Sue
record [Cindy Sue] removed successfully.

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insenistive* |
-----------------------------------------------------------------
choose one: d
===============================

name:    Alice Zue
age:     20
course1: EECS1021       OOP from Sensors to Actuators           MWF   10:30-11:30   ACE 009
course2: EECS4471       Introduction to Virtual Reality         MW    10:00-11:00   CB 122
remarks: ATTENTION! time conflict

name:    Dusan Luc
age:     33
course1: EECS2021       Computer Organization                   MW    17:30-19:00   LAS B
course2: EECS2031       Software Tools                          T     16:30-18:30   SLH E
remarks: SUCCESSFUL! no time conflict

========== 2 records ==========

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one:
```

Implement function **`void remove(struct db_type * pArr[])`** to accomplish the above.

- When the user chooses `P` or `p`, which represents `Swap records,` then starting from the 1st record, swaps the pairs of adjacent records. That is, the 1st record swaps with the 2nd , the 3rd swaps with the

4th ….... If the total number of records is an odd number, then the last record is not swapped with any records.

```
-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: d
===============================

name:    Alice Zue
age:     20
course1: EECS1021       OOP from Sensors to Actuators           MWF   10:30-11:30   ACE 009
course2: EECS4471       Introduction to Virtual Reality         MW    10:00-11:00   CB 122
remarks: ATTENTION! time conflict

name:    Bill Las
age:     21
course1: EECS2001       Intro to the Theory of Computation      MW    14:30-16:00   CLH M
course2: EECS2031       Software Tools                          T     16:30-18:30   SLH E
remarks: SUCCESSFUL! no time conflict

name:    Cindy Sue
age:     33
course1: EECS3101       Design and Analysis of Algorithms       MW    11:30-13:00   SLH C
course2: EECS2031       Software Tools                          T     16:30-18:30   SLH E
remarks: SUCCESSFUL! no time conflict

name:    Dusan Luc
age:     33
course1: EECS2021       Computer Organization                   MW    17:30-19:00   LAS B
course2: EECS2031       Software Tools                          T     16:30-18:30   SLH E
remarks: SUCCESSFUL! no time conflict

========== 4 records ==========

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: p

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: d
===============================

name:    Bill Las
age:     21
course1: EECS2001       Intro to the Theory of Computation      MW    14:30-16:00   CLH M
course2: EECS2031       Software Tools                          T     16:30-18:30   SLH E
remarks: SUCCESSFUL! no time conflict

name:    Alice Zue
```

```
age:     20
course1: EECS1021      OOP from Sensors to Actuators          MWF   10:30-11:30   ACE 009
course2: EECS4471      Introduction to Virtual Reality        MW    10:00-11:00   CB 122
remarks: ATTENTION! time conflict

name:    Dusan Luc
age:     18
course1: EECS2021      Computer Organization                  MW    17:30-19:00   LAS B
course2: EECS2031      Software Tools                         T     16:30-18:30   SLH E
remarks: SUCCESSFUL! no time conflict

name:    Cindy Sue
age:     33
course1: EECS3101      Design and Analysis of Algorithms      MW    11:30-13:00   SLH C
course2: EECS2031      Software Tools                         T     16:30-18:30   SLH E
remarks: SUCCESSFUL! no time conflict

========== 4 records ==========

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one:
```

Implement function **`void swap(struct db_type * pArr[])`** to accomplish the above

Note that efficiency matters. In implementing swap, you should not use strcpy() sprintf() etc to copy/move record data.

- [bouns] When the user chooses `S` or `s`, which represents `Sort database`, the program sorts records based on `age` (in ascending order). Assume the ages are different.

```
-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: d
===============================

name:    Tim Kim
age:     25
course1: EECS3421      Introduction to Database Systems       TR    17:30-19:00   HNE B15
course2: EECS4215      Mobile Communications Networks         TR    17:30-19:00   CB 122
remarks: ATTENTION! time conflict

name:    Alice Zue
age:     20
course1: EECS1021      OOP from Sensors to Actuators          MWF   10:30-11:30   ACE 009
course2: EECS4471      Introduction to Virtual Reality        MW    10:00-11:00   CB 122
remarks: ATTENTION! time conflict

name:    Hysoon Pak
age:     28
course1: EECS3101      Design and Analysis of Algorithms      MW    11:30-13:00   SLH C
course2: EECS3421      Introduction to Database Systems       TR    17:30-19:00   HNE B15
remarks: SUCCESSFUL! no time conflict
```

```
name:    Dusan Luc
age:     18
course1: EECS2011       Fundamentals of Data Structures         TR    13:00-14:30   LSB 106
course2: EECS1001       Research Directions in Computing        W     16:30-17:30   VC 135
remarks: SUCCESSFUL! no time conflict

========== 4 records ==========

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: s

 -----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: d
===============================

name:    Dusan Luc
age:     18
course1: EECS2011       Fundamentals of Data Structures         TR    13:00-14:30   LSB 106
course2: EECS1001       Research Directions in Computing        W     16:30-17:30   VC 135
remarks: SUCCESSFUL! no time conflict

name:    Alice Zue
age:     20
course1: EECS1021       OOP from Sensors to Actuators           MWF   10:30-11:30   ACE 009
course2: EECS4471       Introduction to Virtual Reality         MW    10:00-11:00   CB 122
remarks: ATTENTION! time conflict

name:    Tim Kim
age:     25
course1: EECS3421       Introduction to Database Systems        TR    17:30-19:00   HNE B15
course2: EECS4215       Mobile Communications Networks          TR    17:30-19:00   CB 122
remarks: ATTENTION! time conflict

name:    Hysoon Pak
age:     28
course1: EECS3101       Design and Analysis of Algorithms       MW    11:30-13:00   SLH C
course2: EECS3421       Introduction to Database Systems        TR    17:30-19:00   HNE B15
remarks: SUCCESSFUL! no time conflict

========== 4 records ==========
-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one:
```

[bonus] Implement function `void sort(struct db_type * pArr[])` to accomplish the above
You can implement a sorting algorithm, or explore and use a library function to do the sorting.
In your implementation, you should not use strcpy() sprintf() etc to copy/move record data.

====== FILE IO -- for interested students =======

So far the database is maintained in memory, which means when the program exits, all the records are gone. It would be more useful if we can save the database into the disk, and load them later. To implement this we need disk/file IO, which we may not cover in this course but they may be useful for your future studies.

The following 3 functionalities involve disk IO (implemented for you).

- When the user chooses `W` or `w`, the program writes the current database from memory to the disk file `disk.dat`. If this is the first time writing, the file `disk.dat` will be created in the current directory. Later we can load the record from the disk file.
  If the disk file exists and is not empty (i.e., contains student information written earlier), then the current database information is appended to the disk file. (If the user wants to rewrite the data file, the user can choose `e` or `E` to empty the disk file first (explained below), and then choose `w` or `W` to 'append' the empty disk file.)

- When the user chooses `L` or `l`, loads the database from the disk file `data.dat` which contains some record information that was written earlier. That is, it builds a new database by reading in the disk file. Note that this will overwrite the on-going database (which is in memory). That is, this command will discard the current database and rebuild a new database from the disk file. This is confirmed with the user.
- When the user chooses `E` or `e`, clears the content of the disk file `data.dat`.

  Sample inputs / outputs involving `W/w`, `L/l` and E/e are illustrated below. You are provided with an empty file `disk.dat`.

```
red 537 % a.out

------------------------------------------------------------------
|    (N)ew record         (R)emove record     Swa(p) records     |
|    (S)ort database      (C)lear database    (D)isplay db       |
|    (L)oad disk          (W)rite disk        (E)mpty disk       |
|    (V)iew courses       (Q)uit              *Case Insensitive* |
------------------------------------------------------------------
choose one: d
===============================
========== 0 records ==========

------------------------------------------------------------------
|    (N)ew record         (R)emove record     Swa(p) records     |
|    (S)ort database      (C)lear database    (D)isplay db       |
|    (L)oad disk          (W)rite disk        (E)mpty disk       |
|    (V)iew courses       (Q)uit              *Case Insensitive* |
------------------------------------------------------------------
choose one: n
name: Alice Sue
age: 20
course-1: 1021
course-2: 1001
SUCCESSFUL! no time conflict

------------------------------------------------------------------
|    (N)ew record         (R)emove record     Swa(p) records     |
|    (S)ort database      (C)lear database    (D)isplay db       |
|    (L)oad disk          (W)rite disk        (E)mpty disk       |
|    (V)iew courses       (Q)uit              *Case Insensitive* |
------------------------------------------------------------------
choose one: n
name: Hysoon Pak
```

```
age: 28
course-1: 3101
course-2: 3421
SUCCESSFUL! no time conflict

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: d
===============================

name:    Alice Sue
age:     20
course1: EECS1021       OOP from Sensors to Actuators           MWF   10:30-11:30   ACE 009
course2: EECS1001       Research Directions in Computing        W     16:30-17:30   VC  135
remarks: SUCCESSFUL! no time conflict

name:    Hysoon Pak
age:     28
course1: EECS3101       Design and Analysis of Algorithms       MW    11:30-13:00   SLH C
course2: EECS3421       Introduction to Database Systems        TR    17:30-19:00   HNE B15
remarks: SUCCESSFUL! no time conflict

========== 2 records ==========

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: w

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: c
are you sure to clear db? (y) or (n)? y

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: d
===============================
========== 0 records ==========

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: l
will overwrite current records. are you sure to load disk? (y) or (n)? y
```

```
-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: d
===============================

name:   Alice Sue
age:    20
course1: EECS1021       OOP from Sensors to Actuators           MWF   10:30-11:30   ACE 009
course2: EECS1001       Research Directions in Computing        W     16:30-17:30   VC 135
remarks: SUCCESSFUL! no time conflict

name:   Hysoon Pak
age:    28
course1: EECS3101       Design and Analysis of Algorithms       MW    11:30-13:00   SLH C
course2: EECS3421       Introduction to Database Systems        TR    17:30-19:00   HNE B15
remarks: SUCCESSFUL! no time conflict

========== 2 records ==========

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: q
red 538 % a.out

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: d
===============================
========== 0 records ==========

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: l
will overwrite current records. are you sure to load disk? (y) or (n)? y

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: d
===============================

name:   Alice Sue
age:    20
```

```
course1: EECS1021       OOP from Sensors to Actuators          MWF   10:30-11:30   ACE 009
course2: EECS1001       Research Directions in Computing       W     16:30-17:30   VC 135
remarks: SUCCESSFUL! no time conflict

name:    Hysoon Pak
age:     28
course1: EECS3101       Design and Analysis of Algorithms      MW    11:30-13:00   SLH C
course2: EECS3421       Introduction to Database Systems       TR    17:30-19:00   HNE B15
remarks: SUCCESSFUL! no time conflict

========== 2 records ==========

-----------------------------------------------------------------
|    (N)ew record        (R)emove record     Swa(p) records     |
|    (S)ort database     (C)lear database    (D)isplay db       |
|    (L)oad disk         (W)rite disk        (E)mpty disk       |
|    (V)iew courses      (Q)uit              *Case Insensitive* |
-----------------------------------------------------------------
choose one: q
indigo 539 %
```

Notes

- An executable file named **`sample.out`**, which is my implementation for the assignment, is provided for you to try out different functionalities and their expected outputs. When running this file, you may get "*Permission denied*" error. Then issue command **`chmod 700 sample.out`** to fix this. This file is generated from **`gcc`**, so run it the same way you run `a.out`

- When you download the **`course.txt`** file onto your system, occasionally your system converts it into non-unix format, which will cause malfunction of Sample.out and your code. You can issue **`file course.txt`** to check the format, if it says "*ASCII text, CRLF line terminators*", then issue **`dos2unix course.txt`** to fix this problem.

- Note that **`sample.out`** and **`course.txt`** should be in the same directory. Your **`PE2.c`** should also be in this directory.

- Since the provided code uses `fgets` to read user input. If you use `scanf` in your functions, you may experience some unexpected behaviors -- mixing `scanf` and `fget` has some problems. Thus in your implementation you should try to use `fgets` too. If you really want to use `scanf`, you may need to add a `getchar`() call after `scanf` to consume the extra chars leftover by `scanf`.