Dec 9, 2014: CS teacher suggested I use MySQL for the databases that need to be accessed by the applet and by the offline program

**Criterion B: Design**

The solution will incorporate three databases:

* One for the tutors registered for the program
* One for the tutees registered for the program
* One for the matches made (for the “active” tutors/tutees)

For the offline portion that will take care of managing of the database and attendance:

The offline portion will have 4 classes:

* Tutee
* Tutor
* Match
* TutoringDBMS (database management system)

Classes **Tutee** and **Tutor** have:

* Instance fields (<type> < identifier>):
* String name
* int studentNumber
* char gender
* String email
* String homeroom teacher
* String course
* This will be stored as “ENG1 MPM1” etc, each course identifier separated by spaces
* boolean[] daysFree
* 1 element for each day of the week, true = free
* int attendance
* this is an accumulator
* boolean active
  + set to true if the tutor or tutee is in a match
* Methods:
* Associated methods for **accessing** and **mutating** the above instance fields
* **Write** to Student database

class **Match** (renamed from Pair to accommodate for more than one tutor/tutee) has:

* Instance fields (<type> < identifier>):
* Tutor[] tutor
* Tutee[] tutee
* int numberOfTutors
* int numberOfTutees
* boolean[] meetDays
* days on which the match meets
* String course
* The courses that this match will discuss
* Methods:
* Associated methods for **accessing** and **mutating** the above instance fields
* Method(s) to **email** the tutor(s) and tutee(s) upon a match being made or for an attendance alert
* **Sort** by day of meeting
* **Write** to Match database

class **TutoringDBMS** has:

* Methods:
* main (to be run at the beginning of every session)
* coordinator menu (allows coordinators to make changes)
* add/delete match
* after adding a match, prompt for email/print notification
* review recent registrations (taken from the online applet)
* review day
* informs coordinator if a tutor/tutee has been absent 3 times in a row, prompts for email/print notification (send this to applet?)
* return to “session” operation (see Fig.1)
* exit (done for the day)

Fig.1: Flowchart for operation of main method in class TutoringDBMS:

Main method called

Prompt for coordinator code

If correct

If incorrect

If “session”

If “coordinator menu”

Prompt for day of week (once correct day of week is entered, move on)

Prompt for string

Prompt for “session” or “coordinator menu”

If entered the name of a tutor/tutee that is to be in the session on the specific weekday

Increase the tutor/tutee’s attendance counter

If entered coordinator code

Present coordinator menu

Fig.2: Flowchart for coordinator menu:

“coordinator menu” called

Present following options:

- enter tutor/tutee manually

- add/delete match

- review recent registrations

- review current session

- return to “session” operation

- end session

For add:

- notify if a pair can be made

- specify a specific tutor/tutee for whom to make a match

- set the individuals to “active,” add the match to the Match database

For delete:

- specify which match is to be “deleted”

- set individuals to “inactive,” remove the match from the Match database

Show all recent registrations. If they are deemed valid, add them to the unregistered tutor/tutee database.

Print a table style attendance sheet for the current session.

Return to “prompt for string” in “session” (Fig.1)

Terminate program.

Prompt for information

For the online Java applet portion:

The applet will be used solely for registration purposes by students looking to become a tutor or tutee.

It will ask for the student’s:

* name
* student number
* gender
* grade
* homeroom teacher
* email
* a tutor/tutee option (radio buttons or dropdown menu)
* courses the tutor/tutee would like to tutor/be tutored in (dropdown menu of courses, with a button on the side to “add a course”, which will add a new dropdown menu)
* days free (5 checkboxes for Monday through Friday)

It will store this information in a MySQL database dedicated to registration information. When the offline program is started, the coordinator menu will allow the coordinator to check the registration database and allow the entries to be added to the tutor/tutee database. When an entry is added to the tutor/tutee database, it is removed from the registration information database.