UNLI FRIENDS!
This is an extension of the student record code in the lecture. Instead of using array of structures, you will use DOUBLY LINKED LIST WITH DUMMY to store the records.

1. Add student **[HIGH]**
   ○ Adds a student and stores it in the record list
   ○ Must be sorted by STUDENT NUMBER

2. Add Friends on a Student **[HIGH]**
   ○ Specify a student by inputting student number (student you want to add friends to)
      ■ If student exists specify a student by inputting student number (student to be added to friend list)
         ● If student is existent in list of students, add to the list of friends for that student
         ● If the student does not exist prompt a message that target friends is not in the record then abort
      ■ If the student does not exist prompt an message that target student to add friend to is not in the record then abort
   ○ The friendship is **MUTUAL/SYMMETRIC -** meaning if A is a friend of B then B is a friend of A

3. Edit Student
   ○ Specify student that you want to edit by inputting student number
      ■ If exists, ask the user to edit the name and course
      ■ if not, prompt a message that does not exist then abort

4. Delete Student including Friends **[HIGH]**
   ○ Specify student that you want to delete by inputting student number
      ■ If exists, delete the student in the student list and its list of friends
      ■ if not, prompt a message that does not exist then abort
   ○ You must also delete the instance of this student in all friends list of other students (e.g. If A is a friend of C and A is a friend of D then when A is deleted in the student list A must be deleted in C's friends list and D's friends list)

5. Delete Friend of a Student
   ○ Specify student that you want to delete by inputting student number
      ■ If student exists specify a student by inputting student number (student to be deleted to friend list)
         ● If student is existent in list of students, delete the student to the list of friends for that student
         ● If the student does not exist prompt an message that target friends is not in the record then abort
      ■ If the student does not exist prompt an message that target student to add friend to is not in the record then abort
   ○ The friendship is MUTUAL so it must affect the other student that has the same relationship

6. Delete All Students (Including Friends)
   ○ Delete all records in the list including every student and each of their friends list

7. Delete All Friends of a Student
   ○ Specify student that you want to delete the friends list by inputting student number
      ■ If exists, delete the friends list of that student

- - - if not, prompt a message that does not exist then abort
  - The friendship is mutual so it must affect all student that has the same relationship

8. View Students **[HIGH]**
   - Print the list of students with their student number, name and course
   - (OPTIONAL) You may also print the list of friends of each student

9. View Friends of a Student **[HIGH]**
   - Specify student that you want to print the friends list by inputting student number
     - If exists, print the friends list of that student
     - if not, prompt a message that does not exist then abort

10. Load Data **[HIGH]**
    - Load all data in text file(s) you created from Option 11
    - All data saved must be retrieved and the EXACT state of the save must be properly loaded

11. Save Data [**HIGH**]
    - Save all data in text file(s)
    - You may choose you own format and number of text files to use as long as it works with load (Option 10)

12. Exit **[HIGH]**
    - Exit the program

All Options TAGGED with HIGH must appear and work to get a passing score.

STUDENT DETAILS
1. STUDENT NUMBER
2. NAME (1-WORD required - Multiple Words Optional)
3. DEGREE (1-WORD - Multiple Words Optional)
4. LIST OF FRIENDS

FRIENDS
1. STUDENT NUMBER

LOAD AND SAVE OF DATA
You will load and save data in a text file. You may create 1 singleton file or many text files.The format of the data in the text file is up to you.