COMP 3311 assignment 4
(Implementing a simple information system)
Spring 2015

Due date: 4th May 2015 (12 noon)

Assignment Rules:

1) This is an *individual* programming assignment; you are required to work on your own.
2) The assignment solution you submit must be solely your own work; copying or letting others to copy are both considered cheating.
3) Please run the solution SQL code on Oracle before submitting, if it does not run correctly, you may NOT have the marks.

Assignment description:

You are required to build a simple information system for the University according to the database schema given below, so that the professors can retrieve useful information from the system. You need to complete a program using the C++ language and the ODBC interface. A skeleton of the C++ program is provided to you in the package.

You need to build the tables and insert the records to the Oracle server before you can run the information system. So, you have to log in to Oracle using the SQLPlus client, and run the following three script files, drop_tables.sql, create_tables.sql, insert_records.sql. These three script files are also included in the package.

After running the script files, make sure you issue the “commit;” command at the SQLplus prompt so that the data are physically written to the Oracle

DBMS. You can then start your C++ program. You will be logged in to the Oracle server using *your own Oracle account (*i.e. comp3311stuxxx, xxx is from 001-089). This part has been done for you in the skeleton C++ program provided.

The system should allow a professor to log in with his password and perform queries regarding:

1) Teaching information:

(a) display the course_ID, course_name, offering_no, classroom, and no_of_stds of all the courses he/she is teaching in the current semester (assume the current semester is 'Spring2014'),

(b) display the course_ID, course_name, offering_no, classroom, and no_of_stds of the course he/she is "leading" in the current semester (assume the current semester is 'Spring2014').

(c) group the prerequisites (course_IDs) by the course_IDs of the main courses and display the prerequisites (course_ID) *in a list*. See the screen shot in the *assignment output* section for the expected output.

Hint: you will find the aggregate function LISTAGG() function useful. You can refer to http://www.oracle-developer.net/display.php?id=515 for the exact syntax of LISTAGG().

2) Supervision information:

(a) display all the student_ID, last_name, first_name, phone of all the students he/she supervises.

(b) group the students (student_ID, last_name, first_name) according to the supervisors' staff_IDs, and display the student information *in a list* in ascending order of the student_IDs, see the screen shot for the exact output.

Hint: you may find the LISTAGG() function and the concatenation operator "||" useful.

3) Administrative information:

(a) change the login password for himself/herself,

(b) add a phone number for himself/herself (assume professors do no share phone numbers),

(c) displays the student_ID, last_name, first_name, and phone number for each TA of the course offerings he/she teaches in the current semester (assume the current semester is 'Spring2014'),

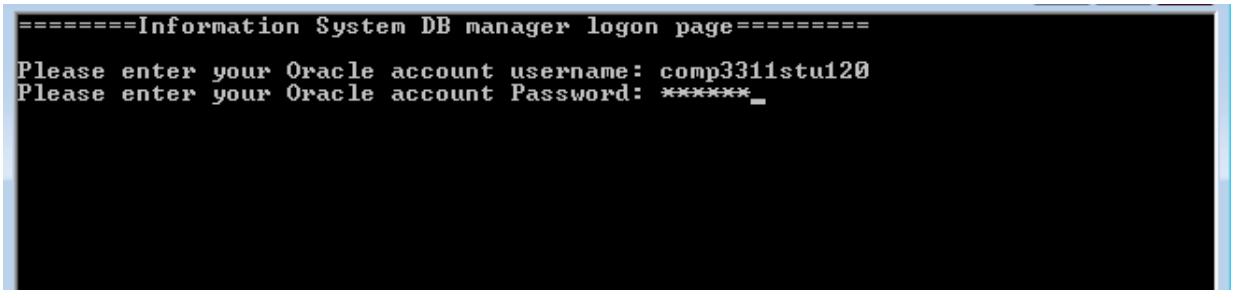
(d) display the lists of *all* the preferred offerings (course_ID, offering_no) for all the TAs, group the result by the TAs' student_IDs. Hint: you will find the aggregate function LISTAGG() function useful.

Assignment output

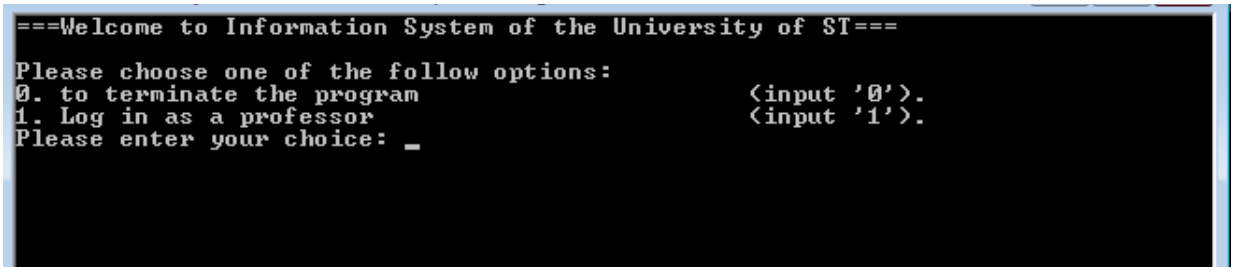
You need to complete the assign4.cpp file based on the provided skeleton file. You just need to fill out the ODBC code for the parts labeled TODO (i.e. the part "TODO: add your code here"). There are altogether nine TODOs. All the nine TODOs involve writing simple SQL statements using the ODBC functions. You can find everything you need in the lab notes of lab 7 and lab 8. For simplicity, you can use the "Direct Execute" method discussed in lab 7 (instead of the "Prepared Statement" method) for the assignment.

To let you focus on SQL/ODBC programming, we assume users always enter valid inputs. You do not need to check the validity of the inputs in the C++ program.

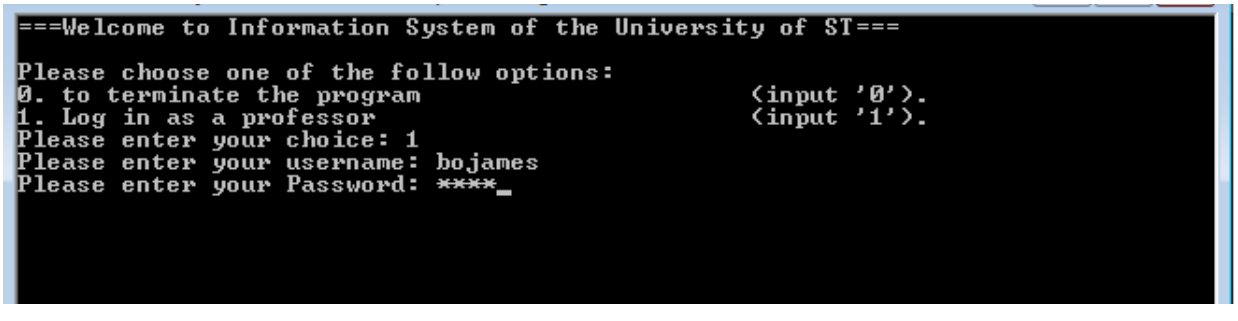
You are the database manager and you need to log in to the information system with your Comp3311 Oracle account (comp3311stuxxx, xxx=001 to 089) before the system is ready to be used by the professors.



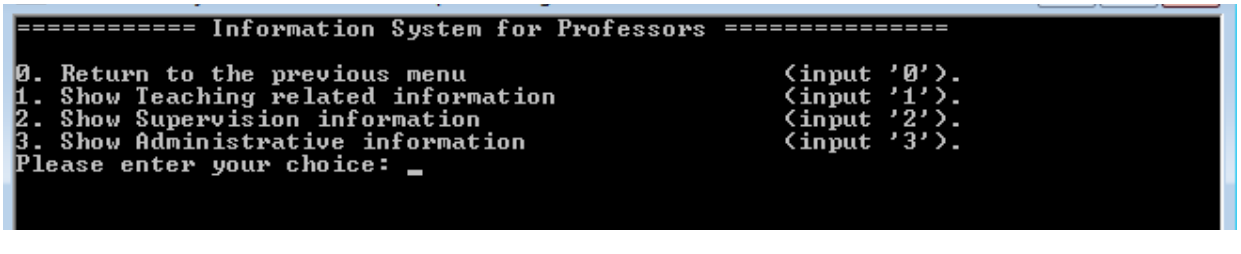
Once the DB manager has logged in successfully, the professors can start to use the information system.



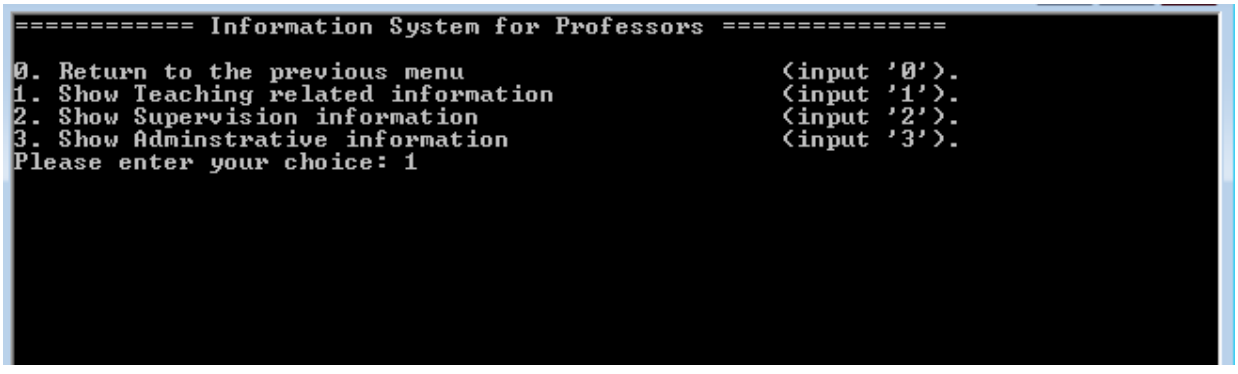
The following illustrates that a professor with the username “bojames” is trying to log in.



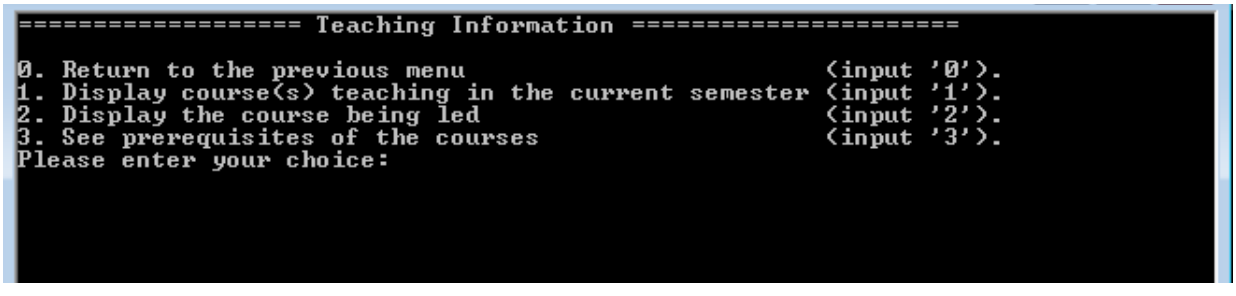
Once he logs in successfully, he will see this main menu of the information system.



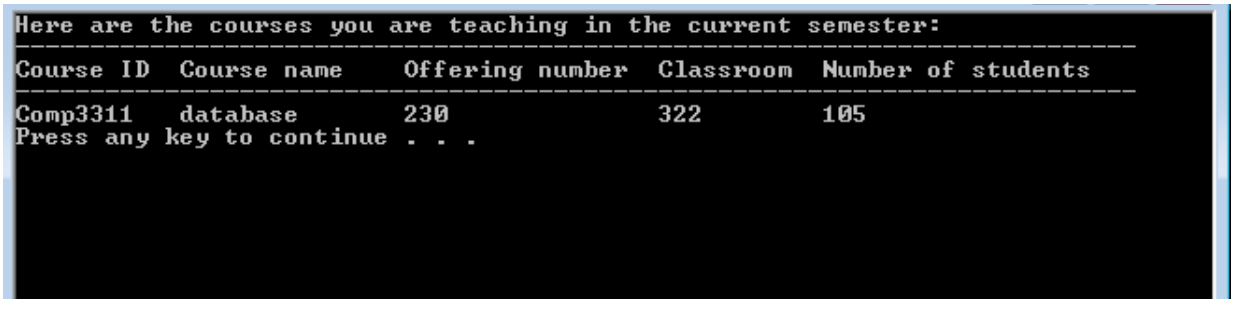
If he selects option “1. Show Teaching related information”,



he will see the following sub-menu.

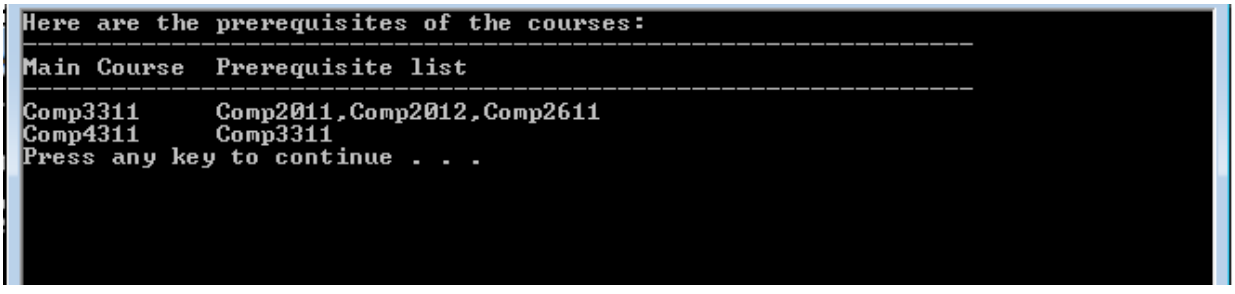


From there he can check the course(s) he is teaching.

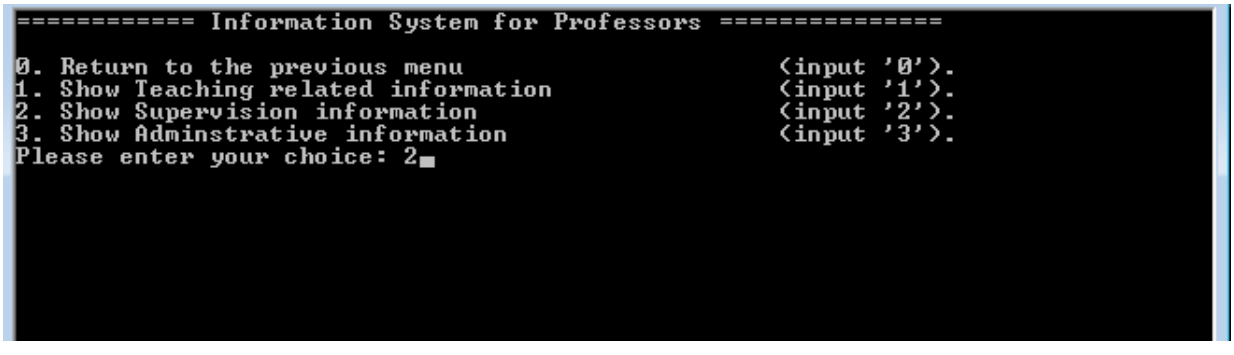


He can check the course he is leading.

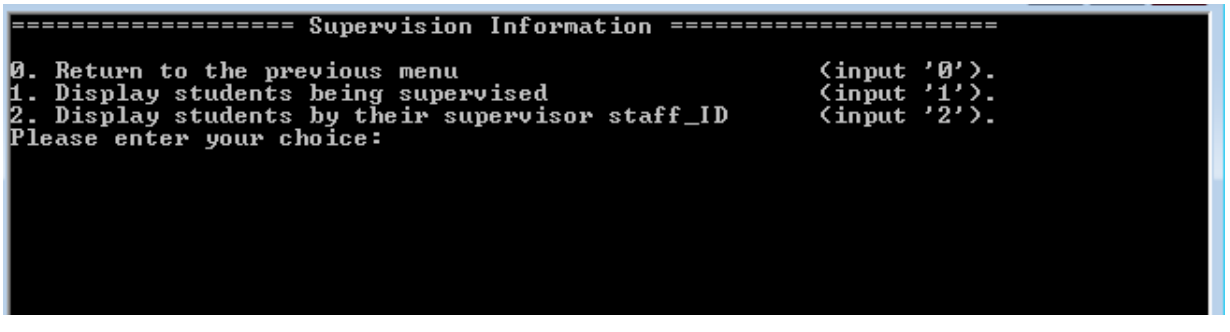
He can also check the prerequisites of all the courses.



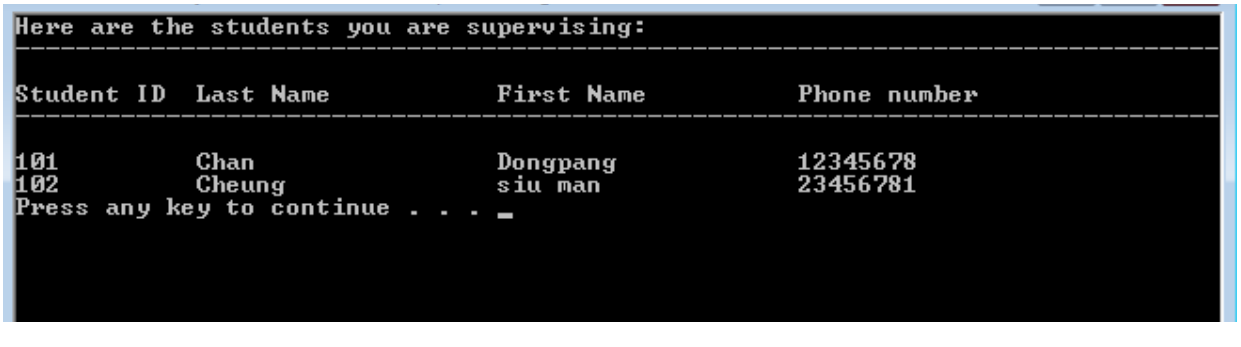
If he selects “2. Show Supervision information from the main menu” from the main menu of the information system,



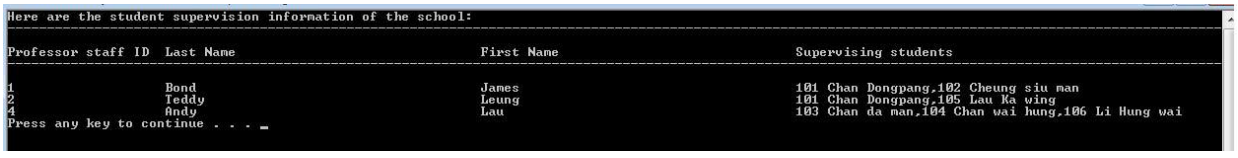
he will see the following sub-menu.



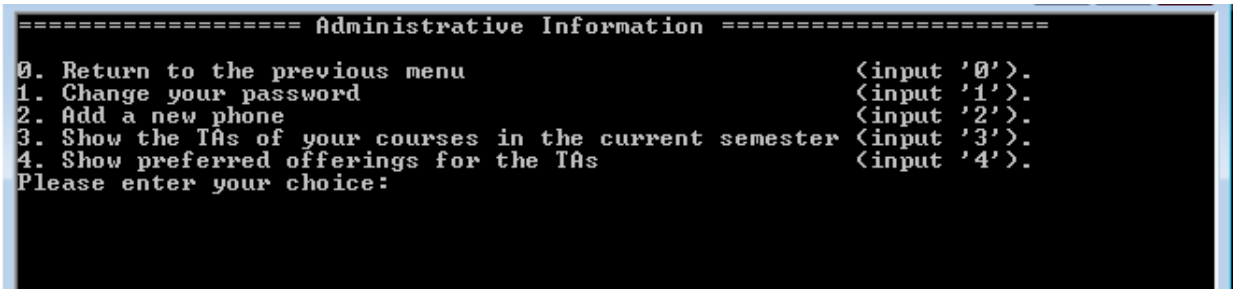
From there he can retrieve all the students he is supervising.



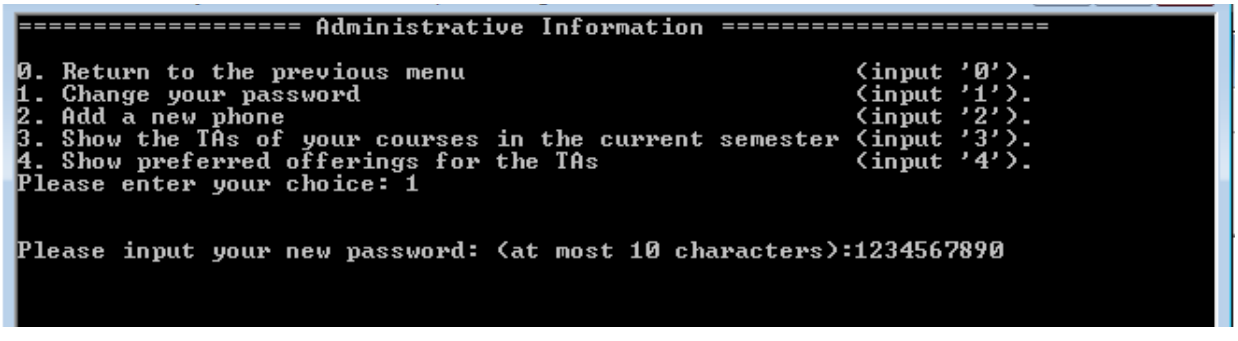
The supervising information of all the other professors.



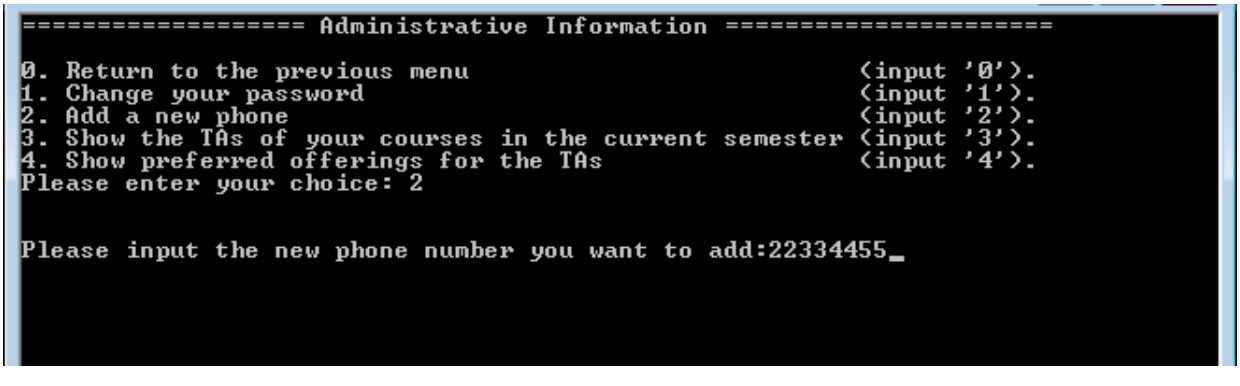
If he selects “3. Show Administrative information” from the main menu, he will see the following sub-menu.



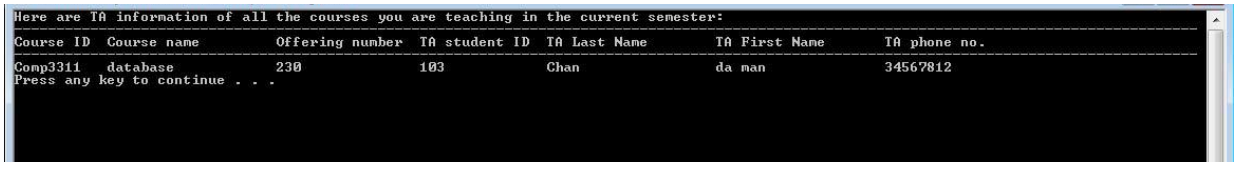
From there he can change his password.



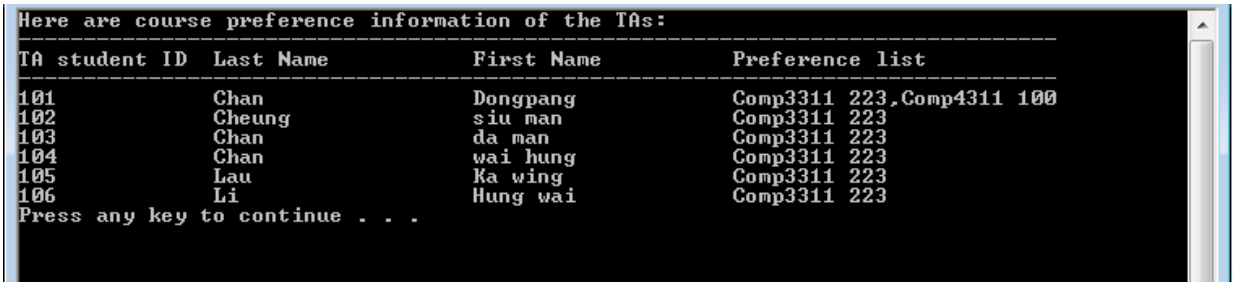
He can add a new phone number.



He can check all the TA(s) that is/are working with him in the current semester.



He can also check the offering preference information of all the TAs for offerings of all the semesters



Disclaimer: The outputs in this section are merely showing some of the instances of the program. Please refer to the executable program provided at the assignment web page for the expected behaviors of the program under all the scenarios.

Further information/reminder about the assignment

1. A skeleton program will be provided to you, you need to complete it by filling all the “TODO” parts. For simplicity you could use the “Direct Execute” method discussed in lab 7 (instead of the “Prepared Statement” method) for the assignment.

2. You may assume the users always enter valid inputs so you do not need to check the correctness of the inputs. You can focus more on the SQL/ODBC codes you need to add.

3. The detailed steps for compiling and running C++/ODBC codes are

available in lab7 and lab8. Read also the appendices of lab7 for setting up the data source and compiling the C++ program under Visual Studio.

4. We will compile and test your codes by using Visual Studio 2010 on the machines in Lab 4210. Refer to the note set of lab7 for the details of how this is done. Make sure your code runs correct on the Lab machines in 4210, otherwise you will not have score for your submission.

Submission

You are required to submit the finished source code “assign4.cpp” to the CASS submission system. Do not submit any other file(s).

We will compile and test your codes by using Visual Studio 2010 on the machines in Lab 4210. If it does not compile or run correctly, no marks will be given.

The link for the submission is:
http://cssystem.cse.ust.hk/home.php?docbase=UGuides/cass&req_url=UGuides/cass/index.html

No Late submission will be accepted!