16.05 Assignment Instructions

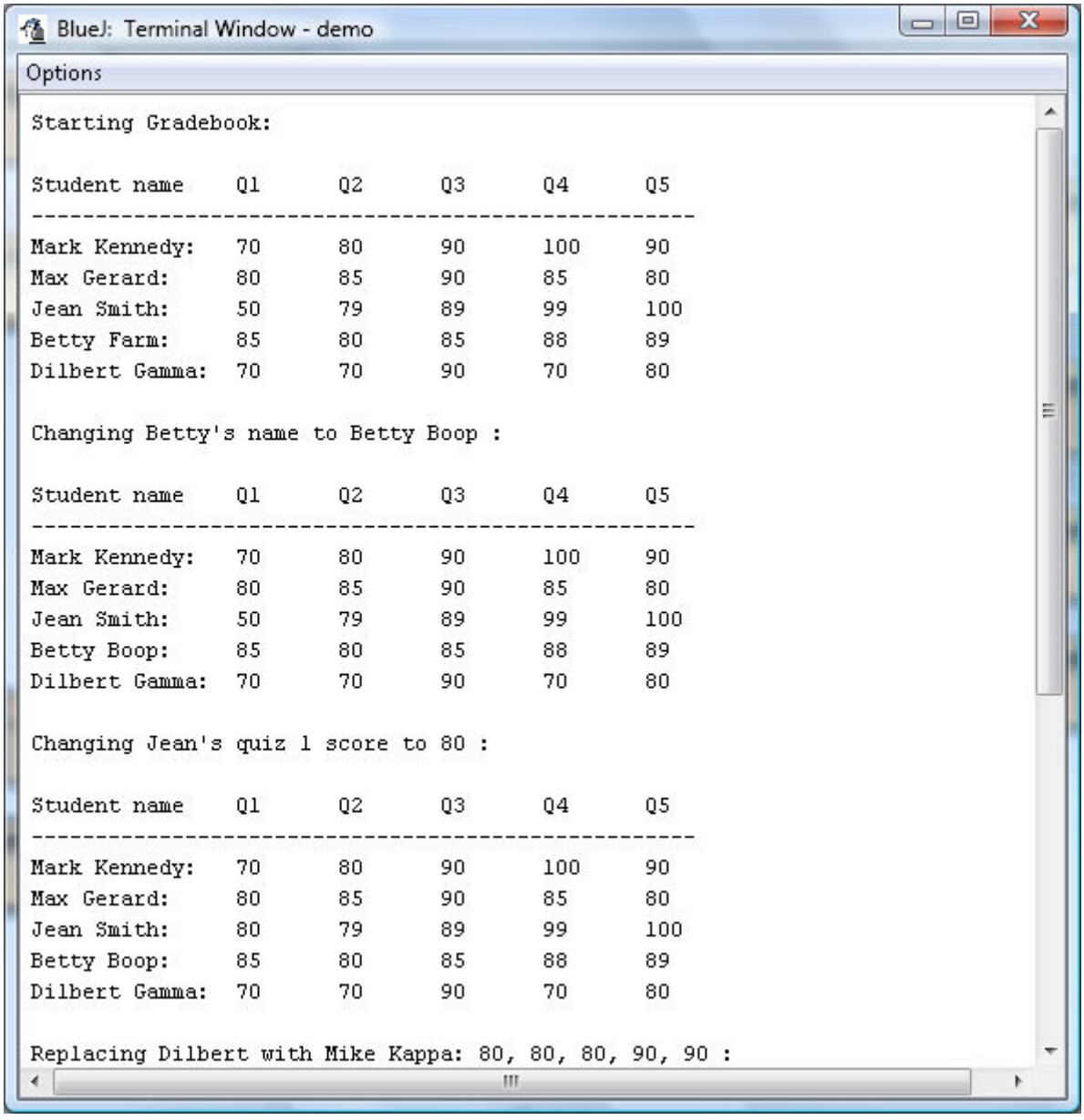
Instructions: For this assignment, you are going to use all your standard algorithm methods.

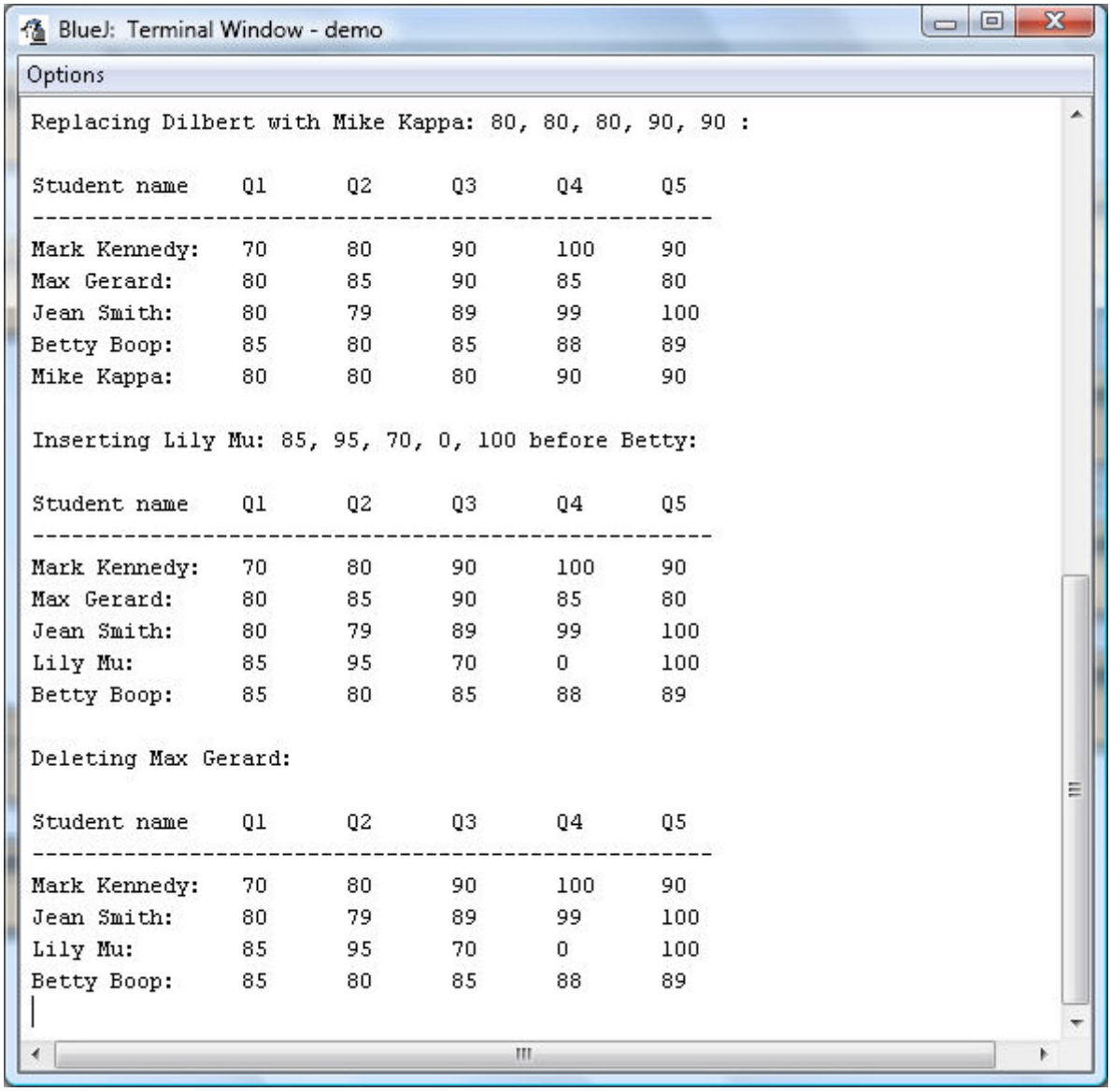
1. Create a folder called **16.05 Assignment** in your module 16 assignments folder.
2. Create a class called **Student**.
    a. Student will need instance variables **`name`** , **`qz1, qz2, qz3, qz4`**, and **`qz5`** (of types **`String`** and **`int`**, respectively).
    b. Student will need appropriate methods and constructors. To make things interesting, create a **`getQuiz()`** method that takes in a quiz number as input and then returns the appropriate quiz value. Likewise, **`setQuiz()`** will take as input a quiz number and quiz score, and then put the value into the right variable. Make sure to have a **`toString()`** method that prints the name of the student along with the quiz scores.
    c. Save the class as **Student.java**.
3. You are to create a class called **`TestStudent`** and save it as **TestStudent.java**.
    a. In this assignment you may choose to use either an array or an ArrayList; there is no need to do two versions of the program.
    b. Make sure that you create data structure called **`myClass`**. Add the following students with their quiz scores.

Candidate	Q1	Q2	Q3	Q4	Q5
Mark Kennedy	70	80	90	100	90
Max Gerard	80	85	90	85	80
Jean Smith	50	79	89	99	100
Betty Farm	85	80	85	88	89
Dilbert Gamma	70	70	90	70	80

    c. Create a method called **`printBook`**() that traverses through the data structure and prints out each element.
    d. Create a method called **`replaceName()`** that replaces a student's name with a new one.
    e. Create a method called **`replaceQuiz()`** that replaces a student's quiz grade with a new one. It should replace only one quiz grade, as indicated, when it is called. It will have the data structure, quiz number, and quiz value as input.
    f. Create a method called **`replaceStudent()`** that replaces a student with another one. It will have the data structure, name to replace, new student name, and quiz scores as input.
    g. Create a method called **`insertStudent()`** that inserts a new student before another student in the data structure. It will have the data structure, name to find, new student name, and quiz scores as input.
    h. Create a method called **`deleteStudent()`** that finds a student by name and then deletes that student.
    i. Remember to make sure your methods handle there being null elements in the data structure.

j. Test your methods. The output should be similar to that shown below:




BlueJ: Terminal Window - demo
Options
Replacing Dilbert with Mike Kappa: 80, 80, 80, 90, 90 :
Student name Q1 Q2 Q3 Q4 Q5
Mark Kennedy: 70 80 90 100 90
Max Gerard: 80 85 90 85 80
Jean Smith: 80 79 89 99 100
Betty Boop: 85 80 85 88 89
Mike Kappa: 80 80 80 90 90
Inserting Lily Mu: 85, 95, 70, 0, 100 before Betty:
Student name Q1 Q2 Q3 Q4 Q5
Mark Kennedy: 70 80 90 100 90
Max Gerard: 80 85 90 85 80
Jean Smith: 80 79 89 99 100
Lily Mu: 85 95 70 0 100
Betty Boop: 85 80 85 88 89
Deleting Max Gerard:
Student name Q1 Q2 Q3 Q4 Q5
Mark Kennedy: 70 80 90 100 90
Jean Smith: 80 79 89 99 100
Lily Mu: 85 95 70 0 100
Betty Boop: 85 80 85 88 89