


King Fahd University of Petroleum & Minerals

Department of Information & Computer Science

ICS 108 – Object-Oriented Programming



ICS108 Project Report

Developed by:

Ahmad Alkhabbaz

201936490

Majed Bamardouf

201918870

Section: 5

Group: 64

Project Description:

The objective of this project is to develop a question bank creator program that supports GUI using JavaFX.

The user of the program can do the following:

- Create multiple choices questions:
 - Each question has the question prompt, a correct answer and 3 wrong choices.
- Edit the questions he created:
 - Can replace the text of question prompt, the correct answer, and the other 3 choices.
- View all the questions he created:
 - Display a page that contains all the questions along with the correct answer and the other 3 choices.
- Delete certain questions:
 - Display a page that contains all the questions, the user can select multiple questions and delete them at once.
- Save all questions upon exit:
 - All questions that have been created will be store in a binary file called “QuestionBank.dat” after closing the program.

Solutions Description:

- The Binary File
- The Questions Class
- Write To The File
- Reading The File
- The Main Menu & the Modify Menu.
- Creating Questions
- Viewing All Questions
- Editing Questions
- Deleting Questions
- Extra feature: Taking the Test

The Binary File

The Binary file “QuestionBank.dat” will contain one array of that contains (Question) objects. To access the file, we used the (ObjectInputStream) and (ObjectOutputStream) classes to read and write the array into the file.

More information about Question class, reading and writing is detailed below.

The Question Class

The question class models multiple choice questions. It is used every time a new question is created. It implements Serializable interface to make it able to be stored in binary files. The UML diagram for it is shown below:

Question	
- question: String	The Question prompt.
- correctAnswer: String	The correct answer.
- otherChoices: String[]	Array of strings for the other 3 choices.
+ Question(question: String, correctAnswer: String, otherChoices: String[])	Create a QuestionObject with the specified parameters.
+ getQuestion(): String	Return the question prompt
+ getCorrectAnswer(): String	Return the correct answer
+ getChoices(): String[]	Return The other choices
+ setQuestion(question: String) : void	Sets a new question prompt
+ setCorrectAnswer(correctAnswer: String) : void	Sets a new correct answer
+ setOtherChoices(correctAnswer: String[]) : void	Sets a new other choices

Reading The File

The file "QuestionBank.dat" is read only once at the beginning of the program. It can be accessed as an array of questions from the questionsArray data field using the readQuestionsFile() method.

The readQuestionsFile() method returns the array of questions from the binary file. to avoid NullPointerException and other runtime errors that may occur in the program, if the file does not exist, the method will create a new file and will write an empty array of Question into it and then return that array.

Writing To The File

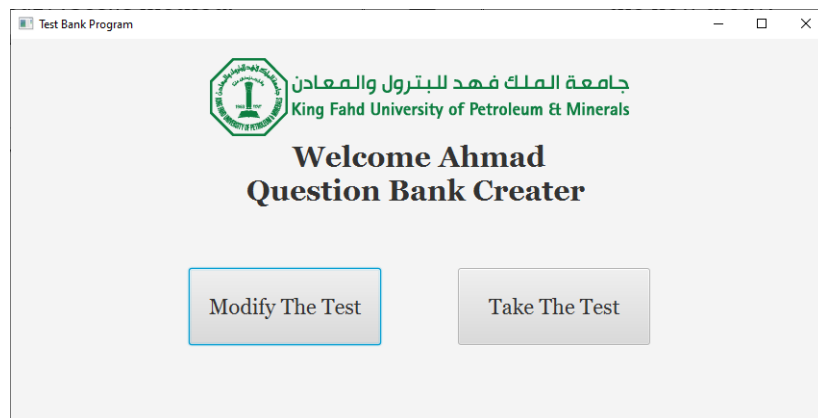
Writing the questions array to the file "QuestionBank.dat" will be done only one time, which is upon exiting the program (after closing the UI). Therefore, writing is done in the main method after launch(args). After that the message "Changes saved to QuestionBank.dat" will be printed on the terminal.

The Main Menu & The Modify Menu

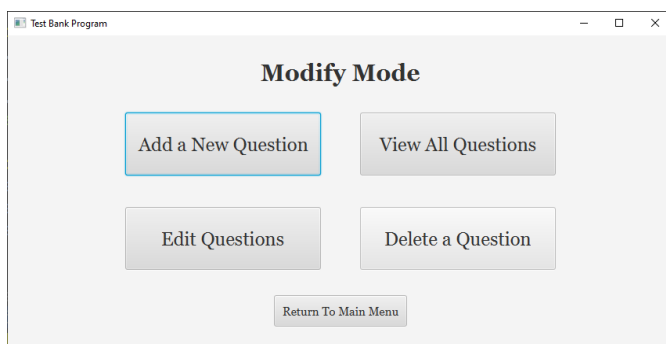
The main menu will be the starting point of the program. It will contain a welcome message to the user that has his computer name (using `System.getProperty("user.name")`). the menu will also contain two buttons. “Modify the Test” button which will lead to modify menu. and “Take The Test” button which is an extra feature to test the questions.

The modify menu is the main feature of this program. It contains four main buttons, add, view, edit, and delete questions. Also it has a button to return to main menu. Each button lead to a menu that has a specific function.

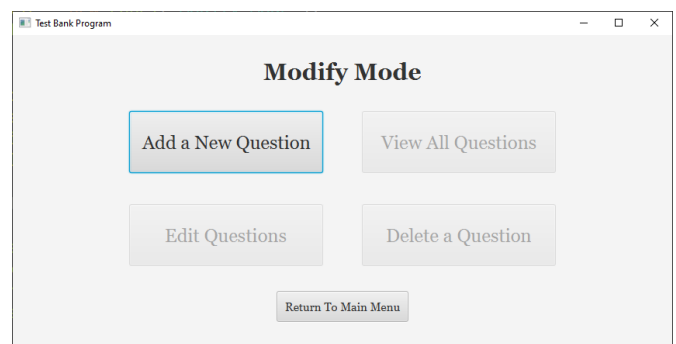
The menu will be displayed by `modifyQuestionsScene` method. The method will also display the corresponding menu of the clicked button. The view, edit, and delete buttons will be disabled if the file contains no question, therefore the user must create a new question to use the other buttons.



The main menu



The modify menu when the file is not empty.



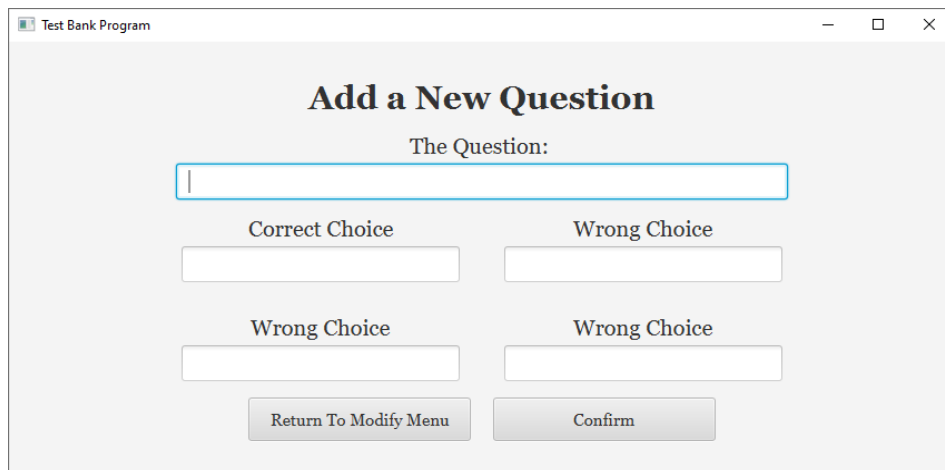
The modify menu when the file is empty.

Creating A Question

To create a new question, the user of the program must go to the “Add a New Question” menu. The scene of this menu will be loaded by createQuestionScene method. This method will display text fields for the user to write the question, correct answer and other choices in them.

After clicking the Confirm button, the button will be disabled, and the method will store the texts in the textfields into a question object, and then add that object to questionsArr using addQuestion method.

addQuestion method will create a new array that has the same questions as the old one (questionsArr), but it will add a new question to the end of it, and then it will replace the questionsArr by the new array.



Test Bank Program

Add a New Question

The Question:

Correct Choice

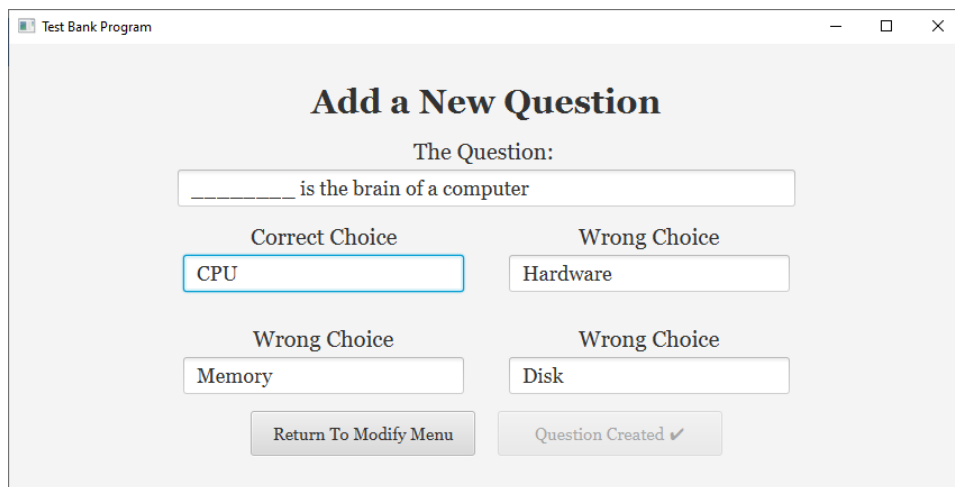
Wrong Choice

Wrong Choice

Wrong Choice

Return To Modify Menu Confirm

Screenshot of (add Question) menu



Test Bank Program

Add a New Question

The Question:

Correct Choice

Wrong Choice

Wrong Choice

Wrong Choice

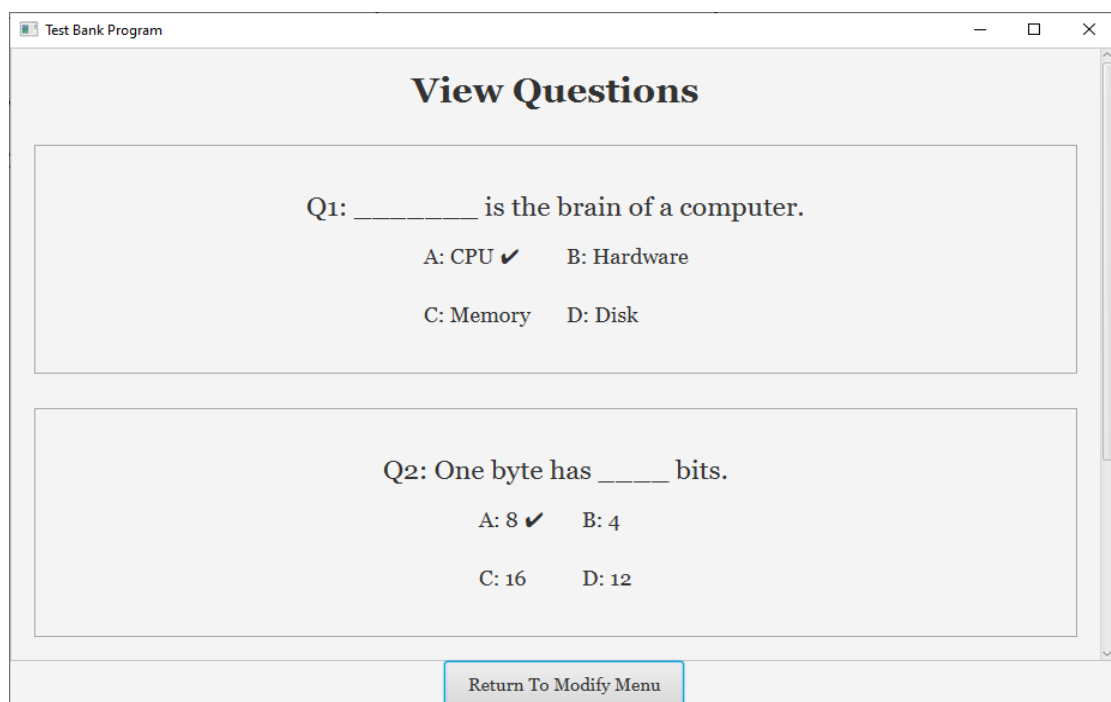
Return To Modify Menu Question Created ✓

The menu after adding a question

Viewing All Questions

After creating questions, the (View All Questions) feature will be available. The menu will display a scroll page that contains every question written in the “QuestionBank.dat” file, as well as the correct answer indicated by check mark (✓), and the other 3 wrong choices.

The method that is responsible for this feature is the `viewQuestionsScene` method. It will traverse through every question object stored in the `questionsArray`. Then it will get its question prompt, the correct answer, and the other choices using the getter methods in the `Question` class, and then place them in appropriate position.



View All Questions menu

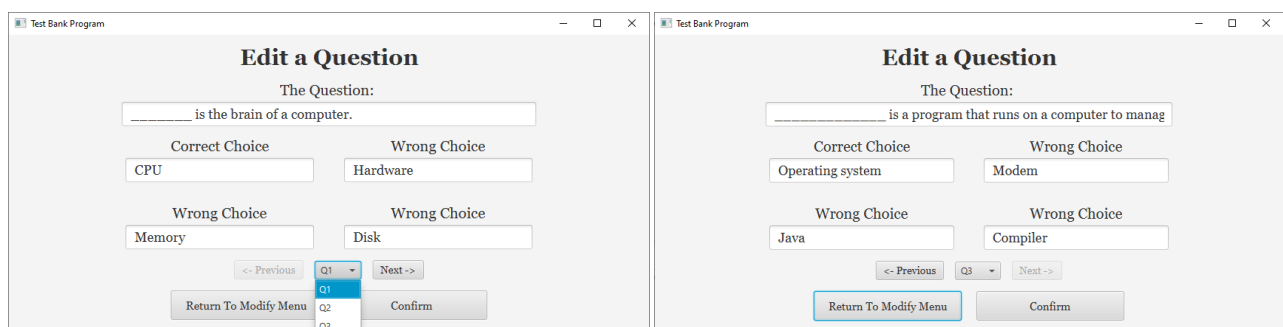
Editing Questions

The user can edit the questions he created from the “Edit Questions” menu. The editQuestionsScene method will display textfields that contains the question and the answers texts for the selected question.

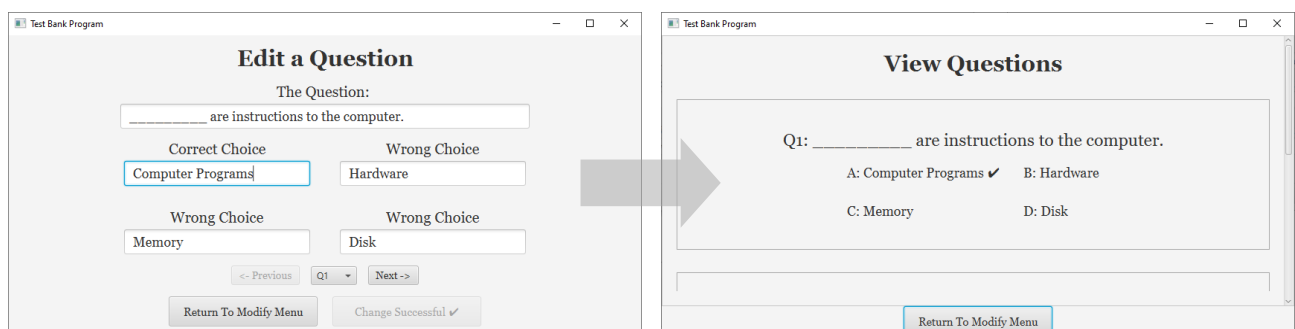
To select another question, there are navigation buttons at the bottom: “Next”, “Previous”, and a combobox that allows picking a specific question. A button will be disabled if there no questions before or after it.

the text for the questions and the answers can be change through the textfields, then click the “Confirm” button to apply the changes to the question object.

The editScene method will handle all the tasks above. It will change the data fields of the question object using the setter methods in the Question class.



Edit questions menu, A button is disabled in each case since there no questions before



Editing the first question

the view question menu shows the question after the change

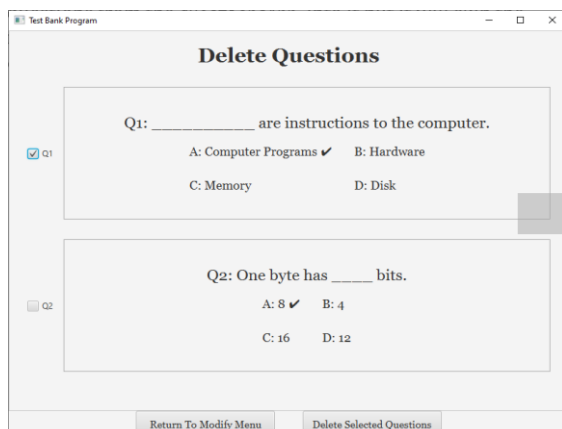
Deleting Questions

The user can delete questions using “Delete Questions” menu. The menu looks similar to “View all Questions” menu but it has checkboxes next to every question.

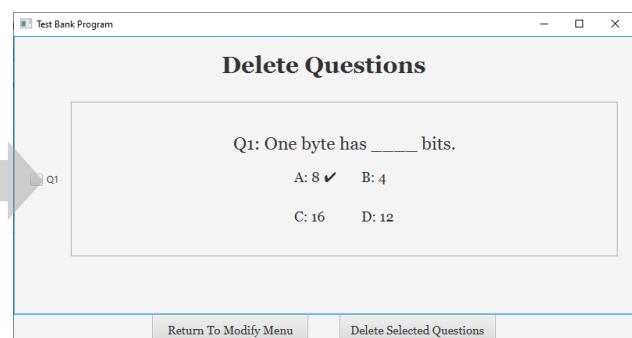
The user can delete multiple questions by selecting them using the checkboxes and then clicking the “Delete Selected Questions” button. The questions is then removed from questionArray using deleteQuestion method.

deleteQuestionsScene method will display the menu for this feature, it is also responsible for searching for the marked question and deleting it using deleteQuestion method. After that, the deleted questions will be removed from the scene.

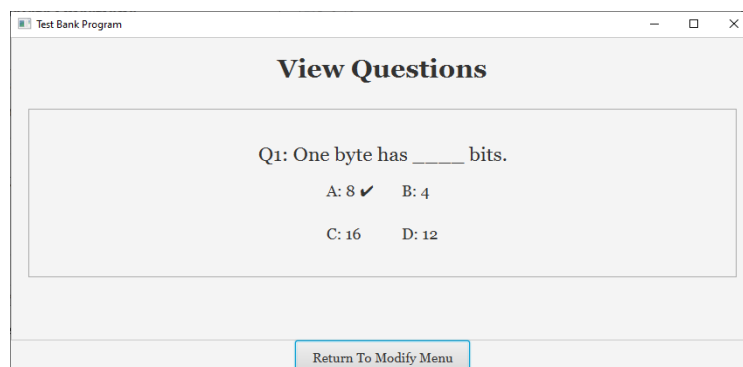
deleteQuestion method will create a new array, it will start adding the old questions from the questionsArr into it but it will skip the question that will be deleted, and then it will replace the questionsArr by the new array.



Deleting the first question



The scene is updated and the deleted question is removed



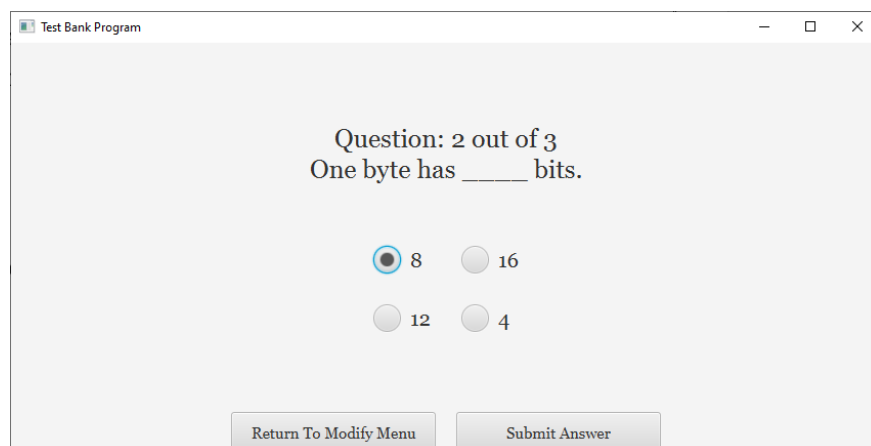
The questions is no longer available

Extra Feature: Taking The Test

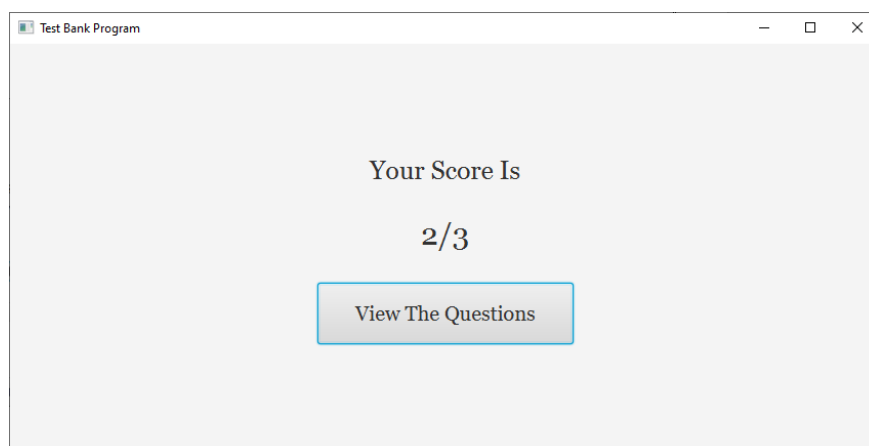
The user can take a quiz that contains the questions he created using the “Take The Test” option in the Main menu. The question order and the answers will be shuffled. All of the operations of this feature is done using the `takeTestScene` method.

`takeTestScene` method will first store the a `arrayList` copy of `questionsArr` name questions and will shuffle it contents. Then it will store the answers of the questions in another `arrayList` named `answers` and shuffle it contents also.

The method then will start displaying the first question and the answers for it. Clicking the submit answer will go display the next question and answers, the process will repeat until the user reaches the end of the test and then his score will appear on the screen. Also the “View The Questions” button will appear so the user can see the correct answers.



Taking the test menu



The score menu

Work Distribution:

The project was done cooperatively. Most of the time one, of us was writing the code and sharing his screen to the other, and vice versa. Except for the report and the TakeTestScene features, which are done separately due to time constraints. Although in the end, both of us verified each others' work.

Both of us were sharing ideas and better solutions to the given problems together.

Therefore the work distribution will be :

50% Ahmad Alkhabbaz
Worked mostly on TakeTestScene
And adding comments

50% Majed Bamardouf
Worked mostly on the report