Chapter 4 System Design

4.1 Overview of Ctutorial4u Architecture

A system development should be based on architecture. CTutorial4u is designed based on rule-based system architecture as depicted in Figure 2.3 in chapter 2 but further enhanced as in Figure 4.1. This is because architecture basically decides on the development technology, hardware and software required to develop the system. At the same time, system design decides on the database design, process and interface flow of the system. Ctutorial4u is basically client-based system. It can be installed in any Windows-based client.

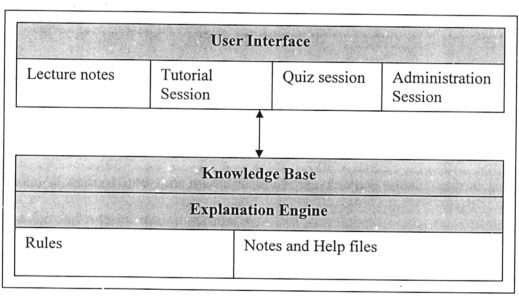


Figure 4.1: Architecture of CTutorial4u

The architecture of CTutorial4u was developed by linking the user interface with the knowledge base and explanation engine. All the rules, static notes and help files are stored in the knowledge base. The tutorial session and quiz session is developed by adding a layer of functions to form the explanation engine. It is supported by a collection of static text files containing explanatory notes, help instructions and rules used by the system which is stored in the knowledge base.

4.2 Process design

4.2.1 System structure chart

System structure design is a chart showing how the system flows from beginning till the end. System flow chart is in hierarchical format. Every hierarchy consists of components called modules. Modules are small unit in a system, which depicts a function. Combination of many modules forms a system. Every function in a system is performed based on the respective modules.

Figure 4.2 shows the structure chart of CTutorial4u. Since, the main intention of the system is to install the system in the laboratory in Faculty of Computer Science and Information Technology of University Malaya, so it designed to have a Login module to prevent it from being used by unintended users of some of the modules such as Quiz session where the administrator (lecturer) decides whom to use the system. The main menu is divided into four modules which are Lecture notes, Tutorial session, Quiz session and Administration session.

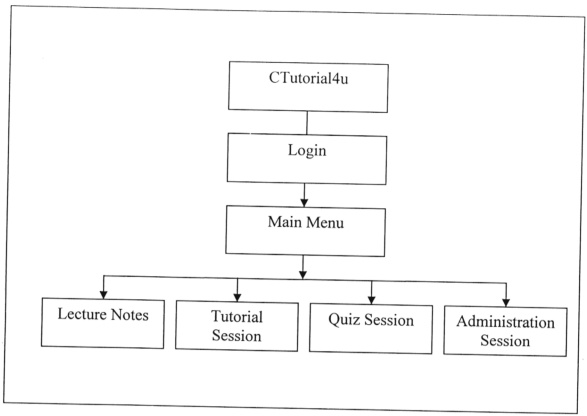


Figure 4.2: Structure chart of CTutorial4u

4.2.2 Flow chart diagram

Flow Chart Diagram graphically characterized data processes and flows in the business system. So, the flow chart diagram is used to describe the system inputs, processes and outputs in the CTutrorail4u.

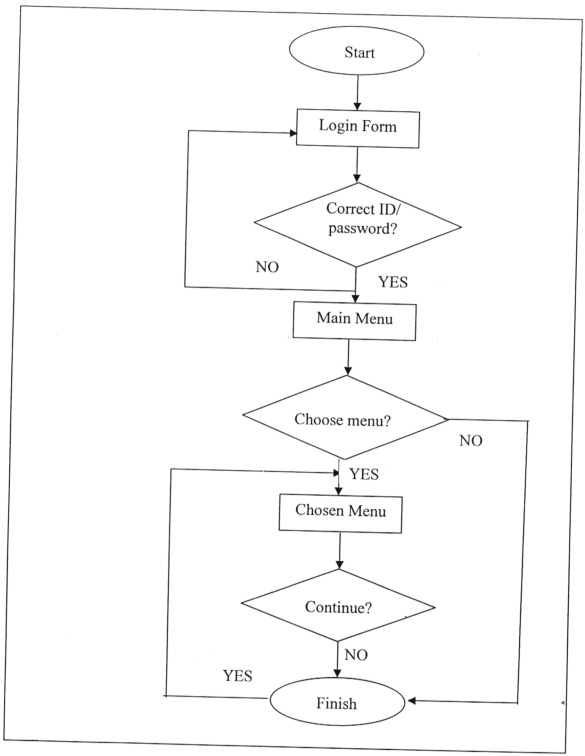


Figure 4.3: Flow chart of CTutorial4u

The Figure 4.3 shows flow chart of the system. Firstly, the system will verify the username and the password supplied by the user and if it doesn't match with the username and password in the database, then a prompt will indicate warning of wrong

username and password won't allow the user to proceed to next screen. If it is correct then the Menu module will be shown for the user to select the four main modules which are lecture notes, tutorial session, quiz session and administration session. If one the menu is selected then the respective screen will be shown and if the user is not keen with continuing with the system the user can opt to exit from the system.

4.3 UML Diagram

Unified Modeling Language (UML) is a language developed by Booch, Jacobson and Rambaugh for specifying, constructing, visualizing and documenting the system and its components [1]. Models can be represented in static or dynamic situations. Each representation has different implications for how the knowledge about the model might be organized and represented.

A static model can be viewed as a snapshot of a system's parameters at rest or at specific point of time. It represents the structural or static aspect of the system. The UML class diagram is an example of a static model. A dynamic model is a collection of procedures or behaviors which taken together to reflect the behavior of a system over time. Dynamic relationships show how the business objects interact to perform tasks. The interaction sequence diagram is an example of UML dynamic model. Firstly, a system can be described by developing its static model, which is the structure of its objects and their relationship to each other at frozen time, a baseline. Then, dynamic models are developed to examine changes to the objects and their relationship over time.

The UML approach adopted for the Ctutorial4u will include the three basic UML graphical diagrams:

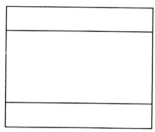
- Class diagram (static model)
- Use-case diagram
- Sequence diagram (dynamic diagram)

4.3.1 Class diagram

There are six main classes in the CTutorial4u such as listed below:

- User
- Admin
- Quiz
- Tutorial
- Explanation
- Results

Table 4.1: Class Notation Table

Component	Explanation
	**Class notation** • Shows a snapshot of the detailed state of the system at a point of time • Rectangle with 3 components. Top compartment represents class name, middle compartment represents its attribute and bottom compartment represents a list of operations.
	Binary associations • Shows associations between two classes. May have an association name.
	**Generalization** • Shows relationship between a more general class and a more specific class.

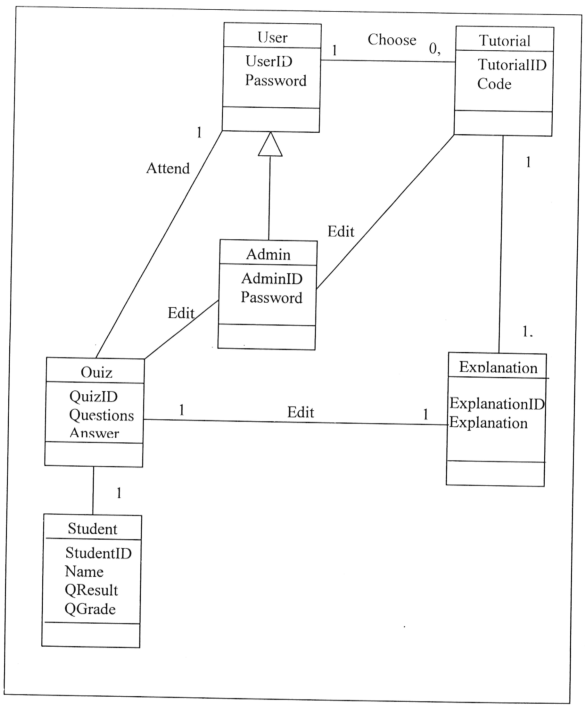


Figure 4.4: Class Diagram of CTutorial4u

The Class diagram depicted in Figure 4.4 shows that 'Users' class is the super class for 'Admin' class. Both the user and administrator are presented in 'User' super class. The diagram mainly depicts the relationship between the user and the classes that associates with it in the Tutorial Mode. Each user has the choice to access the Tutorial Session, hence the (o,*) cardinality. In the Tutorial Session, there are a variety of queries on

different topics with its own unique ID. For each C programming topics there will be various queries and based on the values combination of queries selected the code will be displayed interactively.

The User also can opt to do the quiz session by selecting the intended C programming topics and key in the number of questions to be answered. The questions will be displayed randomly for the user selection of times with timer set 1 minute per question. After the quiz session has ended, the system will generate the percentage of the Quiz for respective users. In times of ambiguity, the user can look up for how and why the Ctutorial4u system came to the possible solution through the Explanation Facility Class. The users can attend many types of quizzes and their results will be stored in Student table.

The administrators from the Admin class are allowed to access all the classes for system maintenance. The associations allow her to edit and modify all the classes easily.

4.3.2 CTutorial4u Use Case Diagram

The use case diagram for CTutorial4u is depicted in the Figure 4.5. The diagram shows the relationship between the actors, the 'User' (Student and Administrator), 'Admin' (Administrator) and use cases.

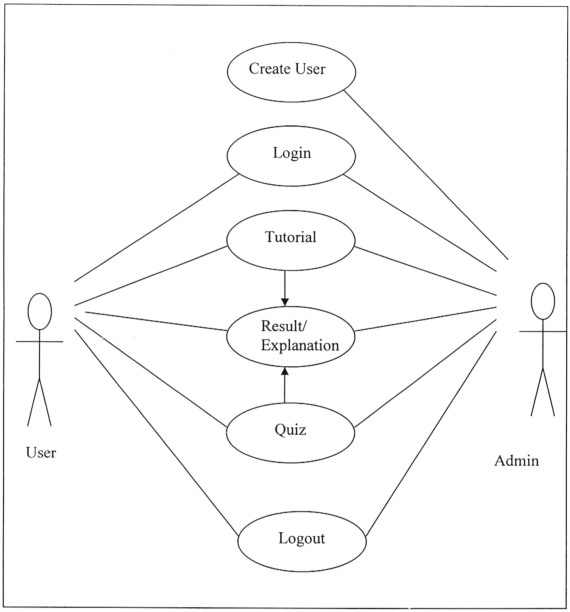
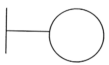
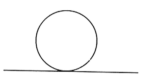


Figure 4.5: Use Case Diagram of CTutorial4u

4.3.3 Sequence Diagram

Sequence diagram shows the interaction between objects in time ordering fashion. It is used in a system to show the interaction between user, screen, objects and entity in the system. It is also to decide the object and class involved in a scenario. Other than that, message series being sent between objects to accomplish scenario function also can be identified.

Table 4.2: Sequence Diagram Notation Table

Component	Explanation
	**Actor** Role of that a user plays in the system, either student or admin
	**Boundary object** Object that an actor and use case associated with. Example of Boundary object is the program interface.
	**Entity objects** Object that contains long-lived information, such as that associated with databases.
	**Control Objects** Object that embodies application logic, used to handle coordination and sequencing.
	Lifeline Line that represents the life and death of an object.
	**Focus of control** Shows period of time which the object is in control of the flow.

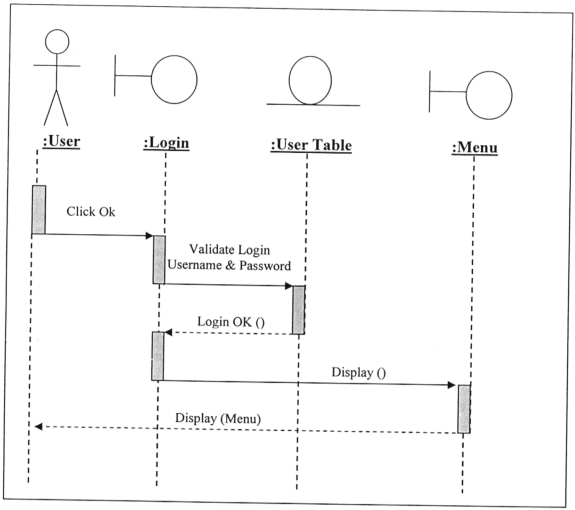


Figure 4.6: Log In Sequence Diagram

The sequence diagram time ordering for the CTuorial4u Log In is as foliows:

I. The first interface of the system is Login form. There will a prompt requesting the user to key in their Username and password.

II. The user enters their Username and password and then click Ok button.

III. The system validates the login information against persistent data in User Account Table.

IV. Then system will display the Menu interface to the User

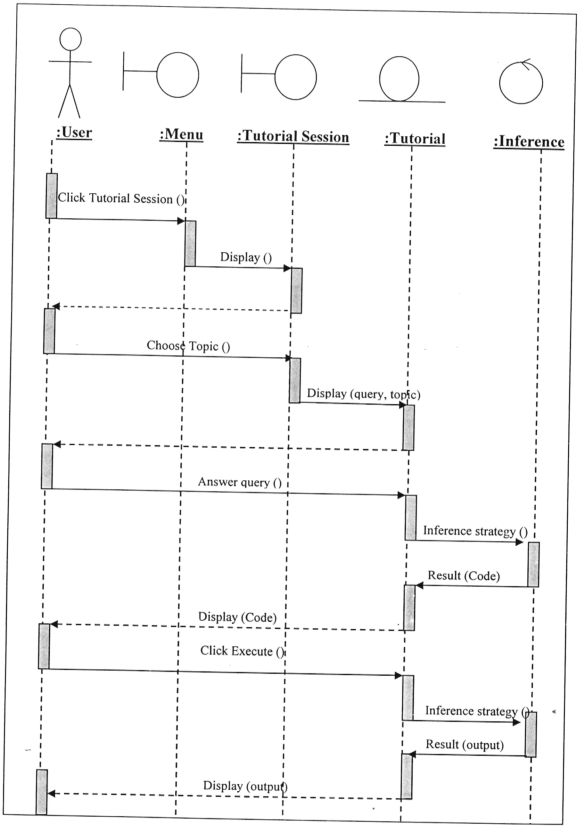


Figure 4.7: Overall Tutorial Session Sequence Diagram

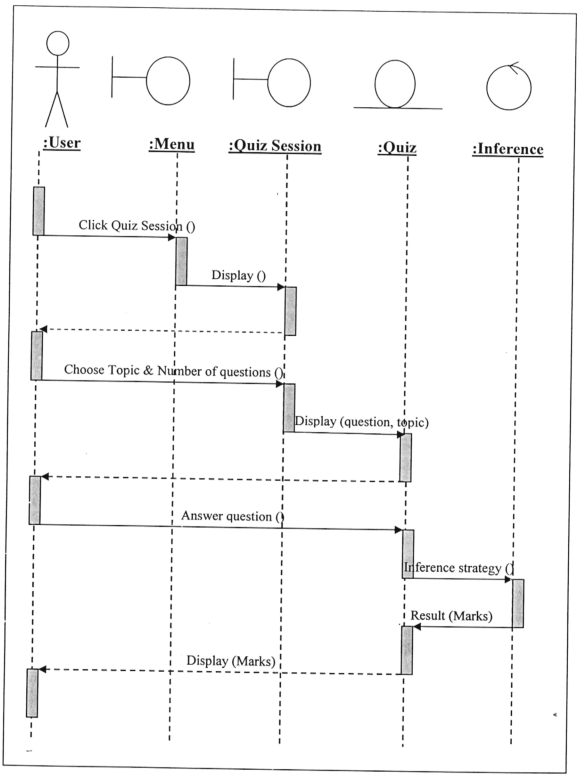


Figure 4.8: Overall Quiz Session Sequence Diagram

The Figure 4.7 shows the sequence of processes in the tutorial session. The sequence is as follow:

I. User click tutorial session button on the menu screen.

II. User interface for the tutorial session will be displayed.

III. Then, the user will select or choose a C programming topic.

IV. Based on the topic, there will be a set of queries will be displayed for the user to answer.

V. The system will display appropriate queries for the selected C programming topic from the knowledge base which is the tutorial table for the user to answer.

VI. Then based on the user's input or answer, the inference strategy will be fired to display the desired program code.

VII. Lastly, the result or the program code will be displayed to the user.

The Figure 4.8 shows the sequence of processes in the quiz session. The sequence is as follow:

I. User click quiz session button on the menu screen.

II. User interface for the quiz session will be displayed to the user.

III. Then, the user will select or choose a C programming topic and the number of questions to answer.

IV. Based on the topic and number of question requested, there will be a set of questions displayed for the user to answer by the quiz table.

V. Then based on the user's input or answer, the inference strategy will be fired to display the result of the quiz.

VI. Lastly, the result or marks of the attempted quiz will be displayed to the user.

4.4 Suggested Database Design

There are few tables created in order to be used in order to process data the CTutorial4u using knowledge base system.

(*) primary key

Table 4.3: User Table

Field Name	Data type	Size	Description
Name	text	50	User's name
Username (*)	text	50	User ID
Password	text	20	Login password
Type	text	20	User type (student/ Administrator)

Table 4.4: Admin Table

Field Name	Data type	Size	Description
Username (*)	text	50	Admin' ID
Password	text	20	Login password

Table 4.5: Tutorial Table

Field Name	Data type	Size	Description
TutorialID (*)	text	50	Tutorial ID
Code	text	255	Program code

Table 4.6: Quiz Table

Field Name	Data type	Size	Description
QID (*)	text	50	Quiz ID
QUE	text	255	Question
A	text	255	Choice A
B	text	255	Choice B
C	text	255	Choice C
D	text	255	Choice D
ANS	text	255	Correct Answer

Table 4.7: Student Table

Field Name	Data type	Size	Description
StudentID(*)	text	50	Student ID
Name	text	100	Student Name
QResult	text	50	Quiz result
QGrade	text	255	Quiz Grade

Table 4.8: Explanation Table

Field Name	Data type	Size	Description
ExplanationID(*)	text	50	Explanation ID
Explanation	memo	3000	Explanation content

4.5 Interface Design

The 'Ten Usability Heuristics' by Jacob Nielsen [8] is taken into account to design CTutorial4u. These are ten general principles for user interface design. They are called "heuristics" because they are more in the nature of rules of thumb than specific usability guidelines.

- **Visibility of system status**

  The system should always keep users informed about what is going on, through appropriate feedback within reasonable time. Every page has their respective name on the top to indicate to the users they are at which page or transaction.

- **Match between system and the real world**

  The system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order. In term of this principle, the system is designed in English language and targeted users are students of Computer Science and Information Technology.

- **User control and freedom**

  Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. CTutorial4u users can exit from the page or system if they don't wish to continue.

- **Consistency and standards**

Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions. All the pages are consistently design where the structure of the navigational bar in every page is designed to be at the bottom.

- **Error prevention**

  Even better than good error messages is a careful design which prevents a problem from occurring in the first place. But at the same time, user's unpredicted actions sometimes require the user to cater for error messages. The Ctutorial4u was designed with high consideration given on how to avoid an error in the system and at the same time if error occurs how the users can accelerate from it.

- **Recognition rather than recall**

  Make objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions to use the system should be visible or easily retrievable whenever appropriate. Ctutorial4u is a simple windows-based system and to help the users to easily understand what they supposed to do next there will be instructions provided at some point of time otherwise the student will easily understand what to do.

- **Flexibility and efficiency of use**

  Accelerators unseen by the novice user may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions. CTutorial4u is designed in such a way that it can be easily used by first year student who do not know what is C programming as a learning tool and also intended to be a reference for the tutors and lecturers.

- **Aesthetic and minimalist design**

  Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility. The warning and error messages are straight forward and simple to understand to avoid the misinterpretation among the users.

- **Help users recognize, diagnose, and recover from errors**

  Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution. As an example, there are error messages or information messages in tutorial session when there is lack of information to process the logic to deliver the program code to the user.

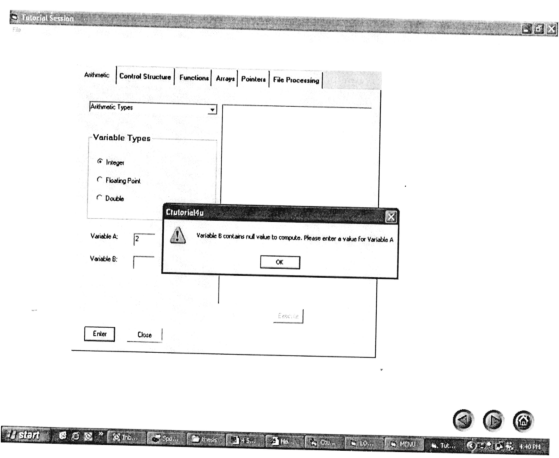


Figure 4.9: Sample of error message in the Tutorial Session

The warning message in Figure 4.9 informs the user that one of the variable field contains null value and suggests what to do to diagnose and recover from the problem.

- **Help and documentation**

  Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large. There is a summary about the CTutorial4u, the contents of the CTutorial4u and technical support who is the person to contact for further information about the system. All these are listed in the menu bar.

4.5.1 Sample User Interfaces

The Figure 4.10 until Figure 4.17 is the sample screen shots of the system. The rest and detailed user interface is captured in the use manual.

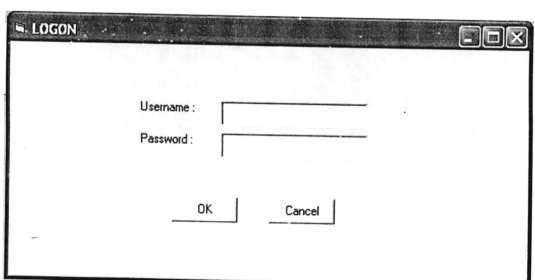


Figure 4.10: Login page

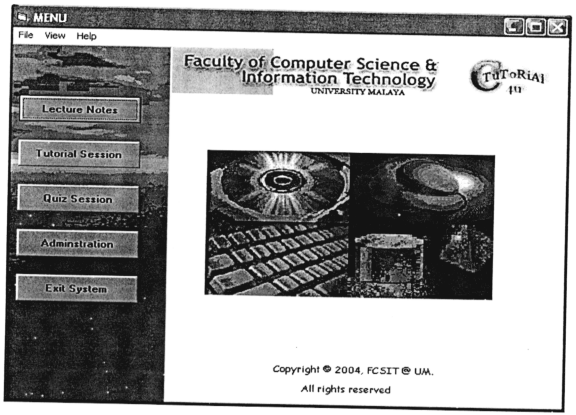


Figure 4.11: Main menu page

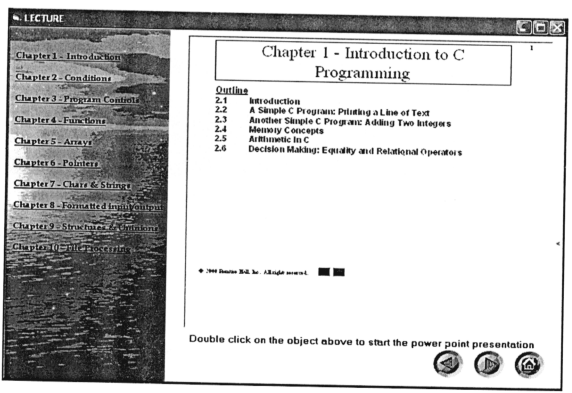


Figure 4.12: Lecture notes page

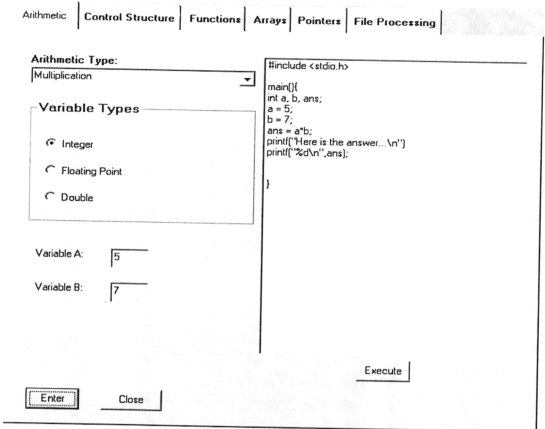


Figure 4.13: Tutorial session page (Arithmetic)

Figure 4.14: Tutorial session page (Arithmetic)

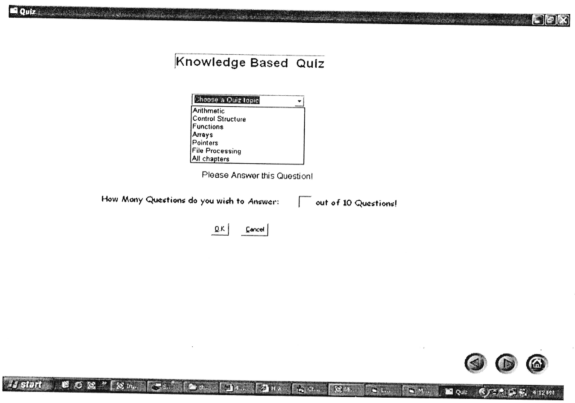


Figure 4.15: Quiz session page

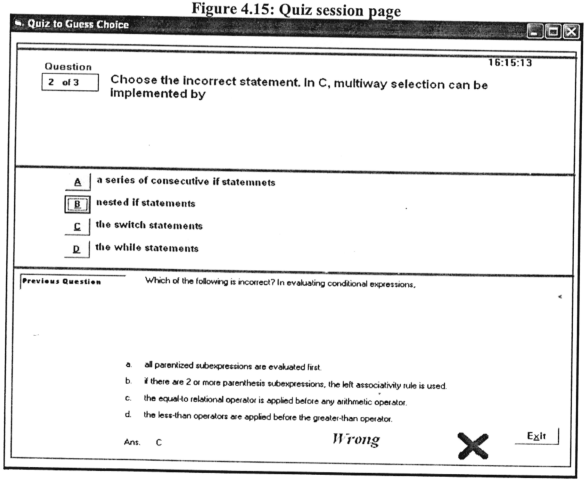


Figure 4.16: Quiz to Guess Choice page

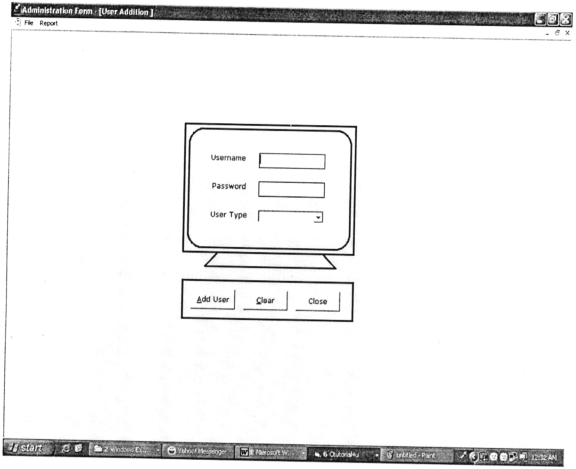


Figure 4.17: Administration session page (Add user)