

# **USER GUIDE**

## **Data Structure Learning Software**

---

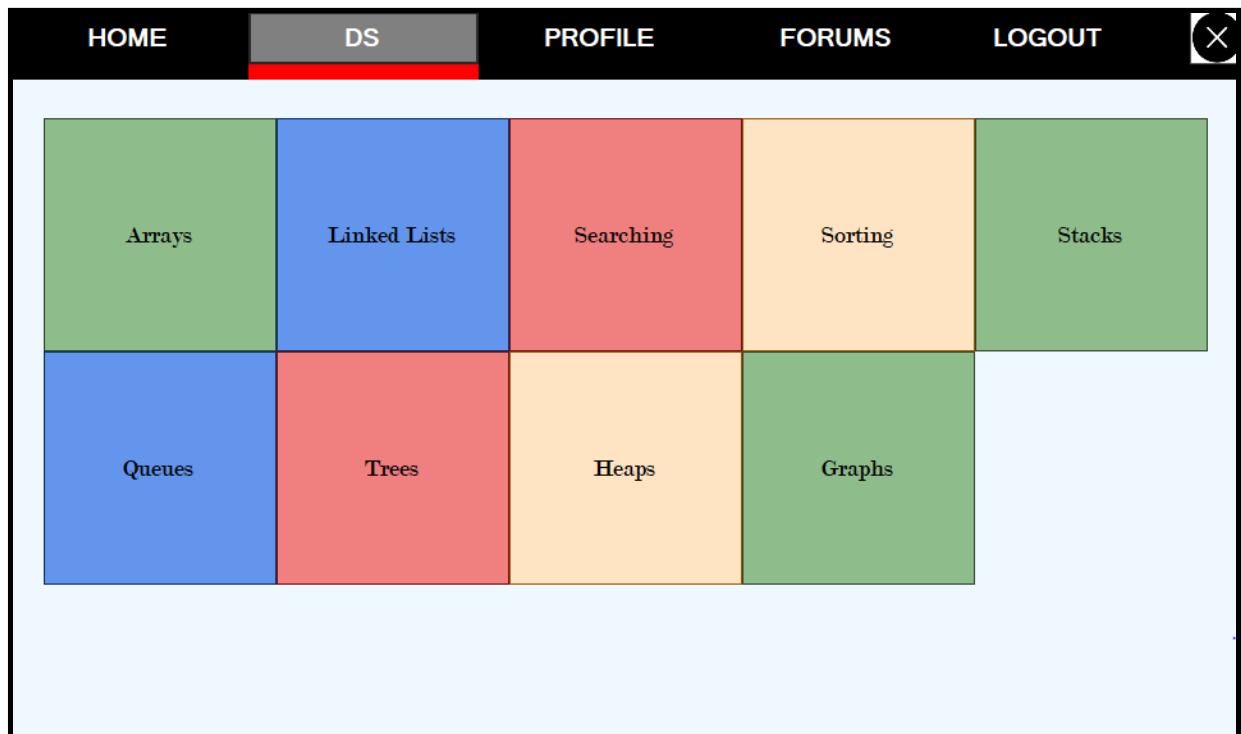
### **Contents**

1. INTRODUCTION	1
2. BASIC FEATURES	2
3. TYPES OF USERS	6
4. LOGIN AND SIGNUP	6
5. MODULES USED	8
6. QUIZZES	10
7. DISCUSSION FORUM	10

# 1. INTRODUCTION

The software is created with an objective to aid the users to understand data structures from their basics. It includes a number of animations, visuals and question to further elevate their understanding and check on their progress.

The software is built assuming that the user has a little or no knowledge regarding data structures. If that isn't the case the user can simply answer some module-wise questions and reach to the desired data structures. Searching, sorting and data structures, namely Array, Linked List, Stack, Queue, Tree, Heap, and Graph, are been covered.



## 2. BASIC FEATURES

The highlighting features of this software application are:-

**Animations for better understanding of the concepts.**

The screenshot shows a web application interface for Selection Sort. On the left, there is a sidebar with a list of sorting algorithms: Bubble Sort, Selection Sort, Insertion Sort, Merge Sort, Quick Sort, and Quiz Time. The main area displays a flowchart for Selection Sort, which starts with an arrow pointing down to a box labeled 'END'. Below the flowchart, there is a section titled 'LEARN WITH FUN' showing an array of numbers: 2, 3, 32, 5, 12, 20, 18. The numbers 2 and 3 are highlighted in pink, and the others are in green. Below the array, there is an input field with the number 18 and an 'Add' button. At the bottom, there are buttons for 'Selection Sort', 'Increase Speed', 'Decrease Speed', 'Pause', and 'Reset'. The top of the interface includes a 'Discussion Forum' button, a 'Welcome, Siddharth' message, and navigation icons.

**Modular division of the materials to ensure a structured learning.**

The screenshot shows a web application interface for Linear Search. On the left, there is a sidebar with a list of search algorithms: Linear Search, Binary Search, Linear Search in Linked List, and Quiz Time. The main area displays the title 'LINEAR SEARCH' and a 'Problem' statement: 'Given an array arr[] of n elements, write a function to search a given element x in arr[]'. Below the problem, there are 'Examples' showing input arrays and search elements, and the corresponding output. A simple approach for linear search is provided, including a list of steps: 'Start from the leftmost element of arr[] and one by one compare x with each element of arr[]', 'If x matches with an element, return the index.', and 'If x doesn't match with any of elements, return -1.' At the bottom, there is a diagram titled 'Linear Search' showing an array of numbers: 10, 50, 30, 70, 80, 60, 20, 90, 40. The number 20 is highlighted in blue, and an arrow points to its index, 6. The top of the interface includes a 'Discussion Forum' button, a 'Welcome, Guest' message, and navigation icons.

Tracking the user's progress to let them start from where they left.

HOME

DS

PROFILE

FORUMS

ADMIN PAGE

✕

# Data Structures Learning Software

Logged in as ADMIN

Array Progress

Linked List Progress

Searching Progress

Sorting Progress

Stack Progress

Queue Progress

A wide range of questions for users to challenge themselves with

Discussion

Welcome, Siddharth

➡

🏠

Introduction

Insertion

Traversal

Search

Quiz Time

```
// as "arr[i]". For example arr1[0] gives us 'g'
// and arr2[3] gives us 40.
```

Usually, an array of characters is called a 'string', whereas an array of ints or floats is called simply an array.

Answer the question correctly to pass the module.

Ques 1. Consider an array A[20, 10], assume 4 words per memory cell and the base address of array A is 100. What is the address of A[11, 5] ? Assume row major storage.

☐ A. 560

☐ B. 565

☐ C. 570

☐ D. 575

Submit

## Discussion forum monitored by teachers for users' doubts.

HOME

DS

PROFILE

FORUMS

ADMIN PAGE

×

☒ Array
 ☐ Linked List
 ☐ Searching
 ☐ Sorting
 ☐ Queue
 ☐ Stack
 

Filt

<b>first comment from lavish</b>	<a href="#">4 comments</a>	Posted 21 Hours ago	By: lavish	admin	Array
<b>ghfj</b>	<a href="#">3 comments</a>	Posted 9 Hours ago	By: SD	as	Array
<b>12345</b>	<a href="#">0 comments</a>	Posted 17 Hours ago	By: SD	as	Array
	<a href="#">0 comments</a>	Posted 17 Hours ago	By: SD	as	Array
<b>5654</b>	<a href="#">0 comments</a>	Posted 17 Hours ago	By: SD	as	Array

## Pseudo codes and codes provided to aid the students to code.

←

Discussion Forum

Welcome, Siddharth

→

🏠

Introduction

Insertion

Deletion

Traversal

Quiz Time

```

/* Given a node prev_node, insert a new node after the given
prev_node */
void insertAfter(struct Node* prev_node, int new_data)
{
    /*1. check if the given prev_node is NULL */
    if (prev_node == NULL)
    {
        printf("the given previous node cannot be NULL");
        return;
    }

    /* 2. allocate new node */
    struct Node* new_node = (struct Node*) malloc(sizeof(struct Node));

    /* 3. put in the data */
    new_node->data = new_data;

    /* 4. Make next of new node as next of prev_node */
    new_node->next = prev_node->next;

    /* 5. move the next of prev_node as new_node */
    prev_node->next = new_node;
}
        
```

Time complexity of insertAfter() is  $O(1)$  as it does constant amount of work.

**Add a node at the end: (6 steps process)**  
 The new node is always added after the last node of the given Linked List. For example if the given Linked List is 5->10->15->20->25 and we add an item 30 at the end, then the Linked List becomes 5->10->15->20->25->30.

## Profile maintained of teachers and students.

ADMIN PANEL

PROFILE

FORUMS

LOGOUT

HOMEPAGE

✕

Username: sid


First Name: Siddharth

Last Name: Agarwal

Designation: admin

Email: shagarwal@yahoo.com

Contact: 1234567890



Change Image

jpg image

Edit Profile

Save

Change Password

Old Password

New Password

Confirm Password

Change

Change Security Question/Answer

▼

Password

Answer

Change

## Module-wise applications and resources for further information.

Introduction

Discussion

Welcome, Siddharth

➔

🏠

Implementation Using Arrays

Implementation Using Linked List

Applications

Play With Stacks

Resources

Quiz Time

Resources for Stacks

Video Lectures: <https://www.youtube.com/watch?v=F1F2imiOJfk>  
Tutorial: [https://www.tutorialspoint.com/data\\_structures\\_algorithms/stack\\_algorithm.htm](https://www.tutorialspoint.com/data_structures_algorithms/stack_algorithm.htm)  
Extra resources of stacks:-  
1. Stack in C++ STL:  
<https://www.geeksforgeeks.org/stack-in-cpp-stl/>  
2. Implement two stacks in an array:  
<https://www.geeksforgeeks.org/implement-two-stacks-in-an-array/>  
3. Stack Class in Java:  
<https://www.geeksforgeeks.org/stack-class-in-java/>  
Standard Problems on Stacks:-  
1. Expression Evaluation:  
<https://www.geeksforgeeks.org/expression-evaluation/>  
2. Infix to Postfix Conversion using Stack  
<https://www.geeksforgeeks.org/stack-set-2-infix-to-postfix/>  
3. Check for balanced parentheses:  
<https://www.geeksforgeeks.org/check-for-balanced-parentheses-in-an-expression/>

5

### 3. TYPES OF USERS

In this software, the user can either work as a guest user or sign-up as either a teacher or a student. The features of these three users vary slightly, as follows:-

#### **Guest User**

Guest user uses the software without signing in and as a result has none of its details stored into the software. Also, progress of the guest users is not stored and they won't be allowed to post their doubts and reply to others' messages.

#### **Students**

Students are signed-in users and hence can exercise all rights and use all facilities of the software. Unlike guest users, they are allowed to allowed to post their doubts and reply to others.

#### **Teachers**

Teachers have all the facilities of user and hence can access the learning materials to study as well. Apart from this, teacher can add questions to the already existing question bank and add, reply and remove a conversation thread as well.

---

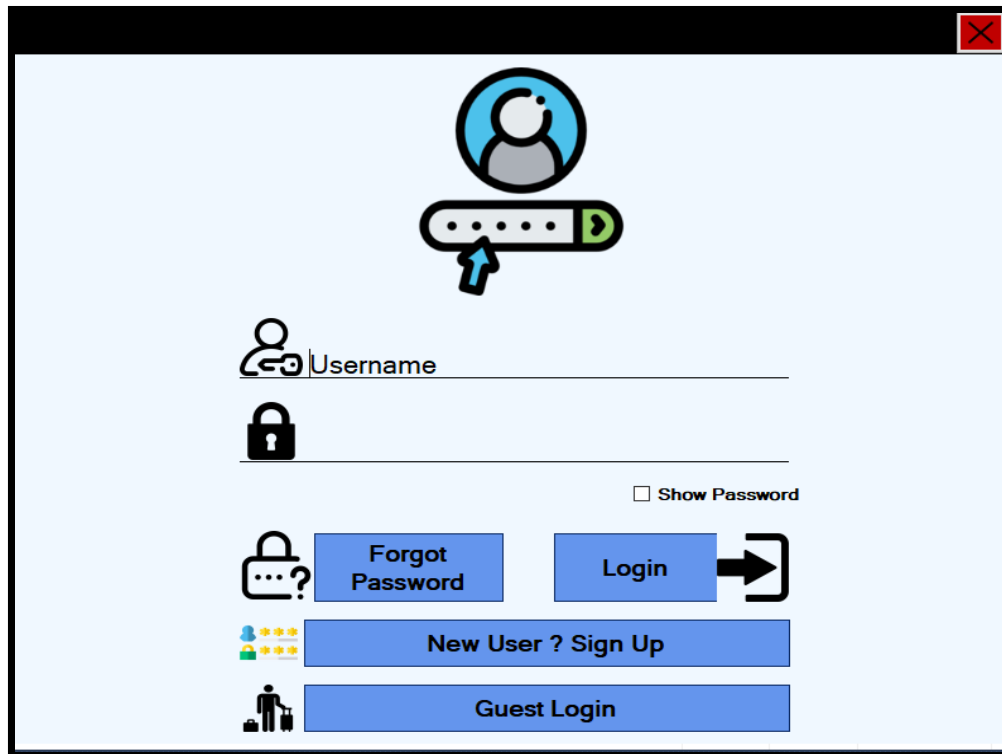
### 4. LOGIN AND SIGNUP

On opening the software the user has an option to login as a guest user, sign into software as an existing user or signing up.

The user can enter his/her credentials to sign in, in which case he will be directed to the home page where his current progress will be shown.

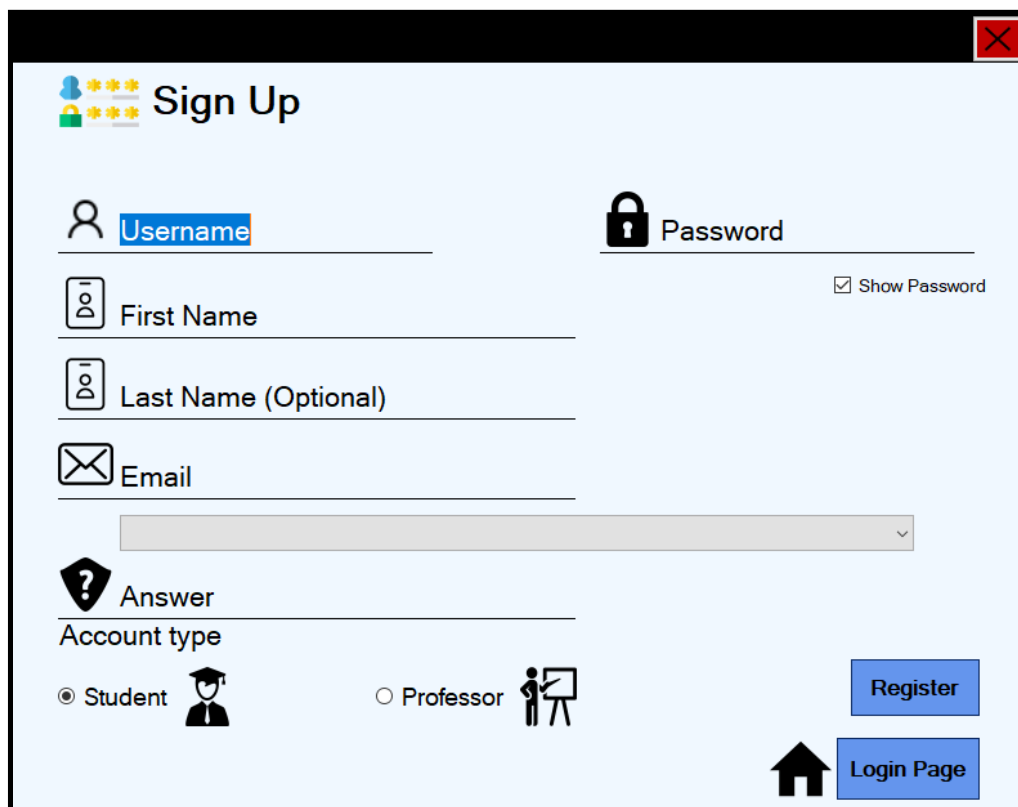
If the user doesn't want to make an account permanently, he/she can login as guest to access the learning material.

Or else the user can make their new account by giving certain details.  
**(refer the images below)**



A login page mockup with a light blue background. At the top center is a large circular icon containing a person silhouette, with a password field (dots and a green eye icon) and a blue arrow pointing to it. Below this are two input fields: the first is labeled 'Username' with a person icon, and the second is labeled with a lock icon. To the right of the password field is a checkbox labeled 'Show Password'. Below the password field are three buttons: 'Forgot Password' with a lock and question mark icon, 'Login' with a right arrow icon, and 'New User ? Sign Up' with a person and stars icon. At the bottom is a 'Guest Login' button with a person and suitcase icon.

## SIGN UP PAGE



A sign up page mockup with a light blue background. At the top left is a 'Sign Up' header with a person and stars icon. Below this are several input fields: 'Username' (with a person icon), 'Password' (with a lock icon and a 'Show Password' checkbox), 'First Name' (with a person icon), 'Last Name (Optional)' (with a person icon), and 'Email' (with an envelope icon). Below the email field is a dropdown menu. Further down is an 'Answer' field with a question mark icon. Below that is an 'Account type' section with two radio buttons: 'Student' (with a student icon) and 'Professor' (with a professor icon). At the bottom right are two buttons: 'Register' and 'Login Page' (with a house icon).



## **5. MODULES COVERED**

### **1. ARRAYS**

One of the most basic concepts of coding, array is the first thing that is taught in this software. In this module, addition, searching and transversal of array is elaborately explained using animations. The animation allows the user to input numbers (maximum of 8) in the array and search or transverse the array.

### **2. LINKED LISTS**

This module deals with introduction to structure and using it to create a linked list. In addition to this, addition, deletion and searching of linked list is also explained using animations. The user can insert numbers into the list and can delete it too!

### **3. STACKS**

After the first two module, the third module deals with stacks and their implementation using the first two. Animations are added to show push and pop operations in both arrays and linked lists that are created by users' inputs..

### **4. QUEUES**

This module is to teach queues and their implementation using both arrays and linked lists. It deals with enqueue, dequeue, front and rear operations in both linear and circular queues. Additionally, animations are added to show the working of above mentioned operations.

### **5. TREES**

This module explains binary trees and operations on it. It deals with addition and deletion operations and explains preorder, inorder and postorder traversal in trees. Additionally, animations are added to show the working of above mentioned operations, users are allowed to give custom inputs for addition and can see the deletion process in the same.

### **6. HEAPS**

This module is to teach heaps and their implementation using arrays. It deals with operations in min-heaps like addition, deletion, extract min etc. Additionally, animations are added to show the working of above mentioned operations.

## 7. GRAPHS

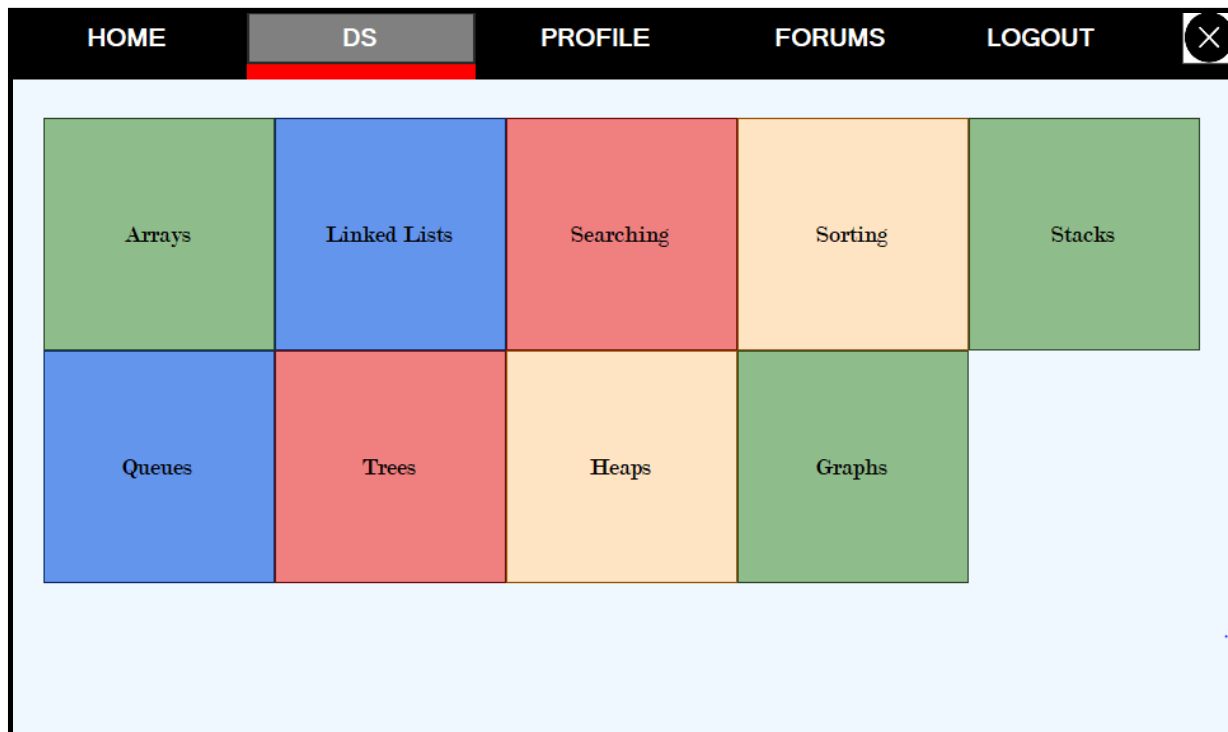
One of the most used data structures, Graphs, are discussed in this module. It explains the graphs, depth first search and breadth first search in the same. Additionally, animations are added to show the working of above mentioned operations.

## 8. SEARCHING

The Searching module shows the measure to search for a particular value in either linked list or array. It explains both linear and binary search and shows animations for the following to explain the methods.

## 9. SORTING

This module elaborates various methods of sorting in increasing order. It explains sorting in both arrays and linked lists. The various methods covered are namely, bubble sort, selection sort, merge sort, bubble sort and quicksort. Moreover, animations are added to show these different methods.

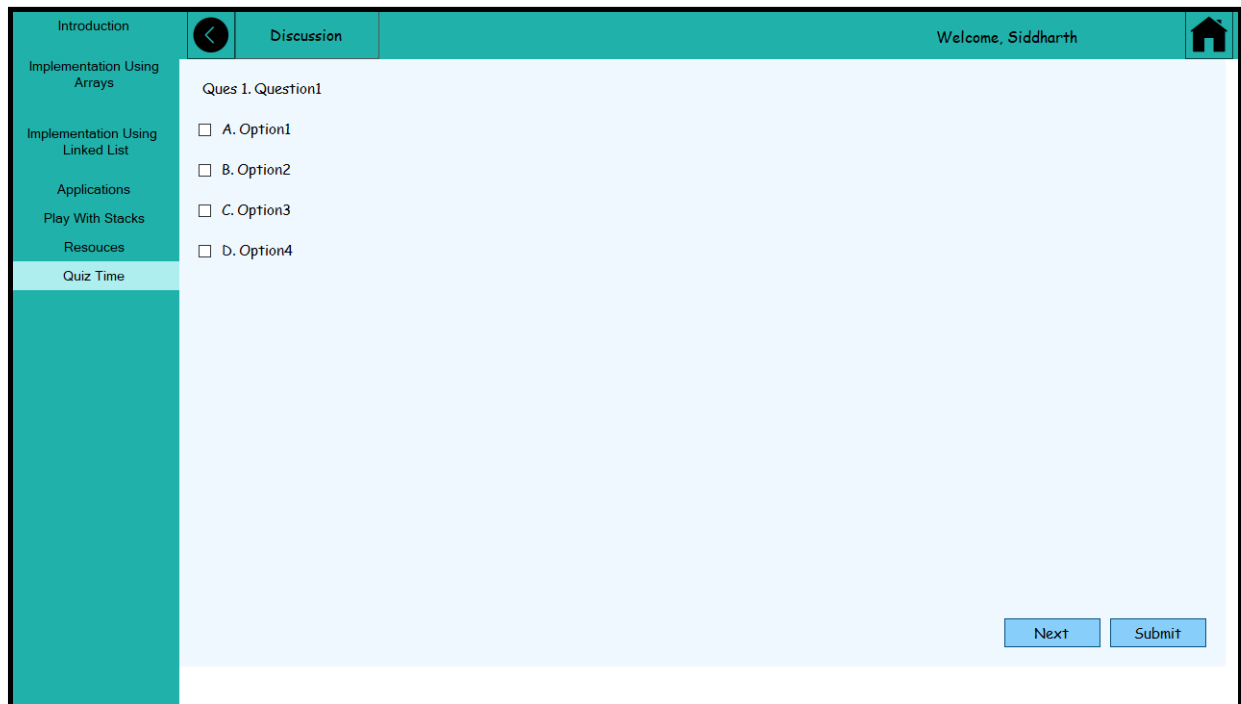


## 6. QUIZZES

Practice is essential for better grasping of a data structure, and hence the data structures learning software picks up random questions to apply the things user has learnt till then.

Every module has topic-wise questions which are to be solved to proceed to the next submodule. These questions are intended to check that user is thorough with the material provided.

Apart from this after completing the module, user can take a quiz to check their proficiency in the topic.



## 7. DISCUSSION FORUMS

The discussion forums are applied in the software to address to the doubts that the user gets during the learning.

The discussion will be sectioned according to the module it is related to, user can assign the same while posting up a message in discussion forum. User can go to discussion forum from their respective modules or from the homepage.

Apart from adding a message, both teachers and students can reply to existing messages, creating a conversation thread. Nesting of replies is not allowed.

In case of any offensive or unrelated message, there is a facility of reporting a message for the users.

The screenshot shows a web interface for a Discussion Forum. On the left is a sidebar with a list of topics: Bubble Sort, Selection Sort, Insertion Sort, Merge Sort, Quick Sort, and Quiz Time. The main area displays a list of forum topics. Each topic entry includes a title, a link to comments, the time since it was posted, the user who posted it, and the user's role. The topics listed are 'first comment from lavish', 'ghfj', '12345', and '5654'. At the bottom right, there is a 'Post' button and a dropdown menu set to 'Array'. The top right corner shows a welcome message 'Welcome, Siddharth' and navigation icons.

This screenshot shows a detailed view of a forum thread. The top section displays the title 'first comment from lavish', the time 'Posted 21 Hours ago', the user 'By: lavish', and the role 'admin'. Below this are links for 'Report', 'Close Thread', and 'Delete Thread'. A section titled 'Replies:' follows, listing four replies from users 'adadaw', 'zdaddd', 'second reply', and 'first reply'. Each reply entry shows the user's name, the time 'Posted 21 Hours ago', and links for 'Report Abuse' and 'Delete Reply'. At the bottom, there is a text input field for replying and a 'Post' button. The sidebar on the left remains the same as in the previous screenshot.