**6. TESTING**

**6.1 Introduction**

Testing is the major quality control measures and during the software development it is used to detect errors that could have occurred during any of the phase like requirement analysis, design, coding. The goal of the testing is to uncover errors in the program.

**6.2 Levels of Testing**

Testing is done in different levels which includes the following.

* Unit Testing
* Integration Testing
* System testing
* Acceptance testing
* **Unit Testing**

In Unit testing each module gets tested during the coding phase itself. The purpose is to exercise the different parts of the module code to detect the coding errors.

* **Integration Testing**

After new testing the modules are gradually integrated into sub systems. It is performed to detect design errors by focusing on testing the interconnection between modules.

* **System Testing**

System is tested against the system requirement if all the requirements are met and if the system performs as specified by the requirement.

* **Acceptance Testing**

It is performed to demonstrate to the client on real life data of the client, the operation of the system.

**6.3 Test Case**

It is the input that tests the genuineness of the program and successful execution Of the test case revels. that there are no errors in the program that are under testing. It is a set of conditions or variables under which tester will determine whether an application or software is working currently

|  |  |
| --- | --- |
| Test case ID | 01 |
| Title | Bubble sort |
| Purpose | To visualize, sort and calculate time and space complexity for array. |
| Test data | Array numbers |
| Steps | 1. Enter array element 2. Click generate 3. Click bubble sort, time and space complexity 4. Display the Sorted output and time and space complexity. |
| Expected  output | The input:  C:\Users\DELL\Pictures\testingimg\Screenshot 2023-06-04 195338.png  The output:  C:\Users\DELL\Pictures\testingimg\Screenshot 2023-06-04 175340.png |
| Test Case ID | 02 |
| Title | Selection sort |
| Purpose | To visualize, sort and calculate time and space complexity for array. |
| Test data | Array number |
| Steps | 1. Enter array 2. Click generate 3. Click selection sort, time and space complexity 4. Display the Sorted output and time and space complexity. |
| Expected  output | The input:  C:\Users\DELL\Pictures\testingimg\Screenshot 2023-06-04 204823.png  The output:  C:\Users\DELL\Pictures\testingimg\Screenshot 2023-06-04 204956.png |
| Test Case ID | 03 |
| Title | Array Deletion |
| Purpose | To visualize and calculate time and space complexity for the deletion of specific element |
| Test data | Array numbers |
| Steps | 1. Enter array and element to be deleted 2. Click generate, delete and time and space complexity 3. If the element is found 4. Element is deleted and displays message with time and space complexity 5. Else displays not found message |
| Expected  output | The input:  C:\Users\DELL\Pictures\testingimg\Screenshot 2023-06-04 212612.png  The Output:  C:\Users\DELL\Pictures\testingimg\Screenshot 2023-06-04 210756.png |
| Test Case ID | 04 |
| Title | Linear search |
| Purpose | To visualize and calculate time and space complexity for the searching of element sequentially from start to end. |
| Test data | Array numbers |
| Steps | 1. Enter array and element to be searched 2. Click generate, serach and time and space complexity 3. If the element is found 4. Displays position of element and time and space complexity 5. Else element not found message |
| Expected  output | The input:  C:\Users\DELL\Pictures\testingimg\Screenshot 2023-06-05 142242.png  The output:  C:\Users\DELL\Pictures\testingimg\Screenshot 2023-06-05 144054.png |
| Test Case ID | 05 |
| Title | Stack operation |
| Purpose | To visualize and calculate time and space complexity for stack push and pop operation |
| Test data | Number |
| Steps | 1. Enter element 2. If stack is empty 3. Display message Underflow 4. Else Element is pushed on to stack 5. If stack is full 6. Display message Overflow |
| Expected  output | The input:  C:\Users\DELL\Pictures\testingimg\Screenshot 2023-06-05 175827.png  The output:  C:\Users\DELL\Pictures\testingimg\Screenshot 2023-06-05 153322.png |

|  |  |
| --- | --- |
| Test Case ID | 06 |
| Title | Queue operation |
| Purpose | To visualize and calculate time and space complexity for queue insertion and deletion at one end |
| Test data | Number |
| Steps | 1. Enter number 2. If Queue is filled 3. Display Overflow 4. Else Insert at rear end of Queue 5. If Queue is empty 6. Display Underflow 7. Click on the time and space complexity 8. Displays the time and space complexity |
| Expected  output | The input:  C:\Users\DELL\Pictures\testingimg\Screenshot 2023-06-05 174115.png  The output:  C:\Users\DELL\Pictures\testingimg\Screenshot 2023-06-05 174303.png |

|  |  |
| --- | --- |
| Test Case ID | 07 |
| Title | Singly linked list |
| Purpose | To visualize and calculate time and space complexity forinsetion at the beginning of linkedlist. |
| Test data | Node element |
| Steps | 1.Enter the node element and index  2.Insert the node at index set its link to nextnode  3.Enter the node element or index to be deleted  4. If found node element or index is deleted  5. Else Display error message |
| Expected  output | The input:  C:\Users\DELL\Pictures\testingimg\Screenshot 2023-06-06 105312.png  The output:  C:\Users\DELL\Pictures\testingimg\Screenshot 2023-06-06 120549.png |
| Test Case ID | 08 |
| Title | Tree |
| Purpose | To visualize and calculate time and space complexity forinsetion at the beginning of linkedlist. |
| Test data | Node element |
| Steps | 1.Enter the node element  2.Click on the create BST  3.Displays the BST  4. Click on the pre-order  5. Displays the pre-order list |
| Expected  output | The input:  C:\Users\DELL\Pictures\testingimg\Screenshot 2023-06-07 151407.png  The output:  C:\Users\DELL\Pictures\testingimg\Screenshot 2023-06-07 152635.png |
| Test Case ID | 09 |
| Title | Graph |
| Purpose | To test the path exists from source to all other nodes. |
| Test data | Node element |
| Steps | 1. Enter the number of vertex, matrix elements, source vertex.  2.Click on the Run-algorithm  3.Displays the vertex sequence  4.Click on the time and space complexity  5.Displays the time and space complexity |
| Expected  output | The input:  C:\Users\DELL\Pictures\testingimg\Screenshot 2023-06-07 154936.png  The output:  C:\Users\DELL\Pictures\testingimg\Screenshot 2023-06-07 145104.png |