International Association of Scientific Innovation and Research (IASIR)
(An Association Unifying the Sciences, Engineering, and Applied Research)

ISSN (Print): 2279-0063
ISSN (Online): 2279-0071

International Journal of Software and Web Sciences (IJSWS)
www.iasir.net

Visualization Tool for Tree and Graph Algorithms with Audio Comments

[1]Urvashi Pathania, [2]Aman Singh
Dept. of Computer Science and Engineering,
Lovely Professional University, Punjab, India

Abstract: *Data structure is mathematical and logical representation of data in memory. Now the era of this time online learning or and e-learning are widely used in higher education, every student wants flexibility in learning environment that's why visualization tool come in fashion. Visualization tool is graphical representation of data in static or dynamic 2D or 3D format. In this paper we present visualization tool of tree and graph of data structure integrated with audio comments. Using this tool student solve many problem that they faced when the execute algorithms of tree and graph and also do self study at home because of online availability of tool.*

Keywords: *visualization tool, audio comments, visualization tool of tree and graph, visualization with audio comments, tree and graph algorithms.*

I Introduction

In computer science engineering data structure is one of the important subject for undergraduate students but still student find difficulties to understand data structure, its operation and its algorithms due to its complex nature. Data structure is mathematical and logical representation of data. Data structure basically categorized in two parts i.e. linear data structures and non linear data structures. In linear data structures we have arrays, queue, stack and linked list and in non linear data structures we have tree and graph. In this paper we implement non linear data structures algorithms because most of students find difficulties when they worked with different algorithms of trees and graph data structures.

Visualization tool is one of the best tool now days used by students and educator of many university to solve the problem of algorithms some extant but still visualization tools have some problem they only visualized the algorithms but they don't illustrate what they are visualized. In this paper we implement visualization tool of tree and graph algorithms of data structure integrated with audio.

II Literature Survey

Robert Meolic (2013) et al. in this paper author discussed how they implement mobile framework of sorting algorithms. In this implement three sorting algorithms they are insertion sort, quick sort and Gnome Sort. For implementation they used c- style code. Disadvantage of this framework is that they are restricted to some algorithms.[1]

Fuad Alhosban (2012) et al., in this paper author implement an interactive visualization tools named DSL in which visualization is integrated with audio. To develop this tool we used java. The DSL (data structure learning) tool, basically have three parts named: Basic Objects, Nodes, and Integration of audio with visualization. [2]

Crescenzi Pilu, Malizia Alessio, Verri M. Cecilia, Díaz Paloma and Aedo Ignacio (2012), In this paper author implement the full integration of algorithm visualization movies with the lessons material of data structure. In this students measured that AV movies is useful tool for individual f study. For developing AV movies, Algorithm visualization system is used called AlViE, which is a post-mortem tool. This tool divided into three parts they are (1) a visualization player, (2) a graphical input developer that allows the user to create new input (3) a Java class library. [3]

Anghel Traian, Florea Adrian, Gellert Arpad (2011), In this paper author discussed online learning system, Education Management Tool (EDM), the basic agenda of this tool is to improve consistent coaching style and examination methods, without replacing them. This tool has three parts e-learning, e-testing, e-management. The technology used for development is ASP.NET (.aspx), combined with AJAX javascript technologies and for database Microsoft Sql Server 2008. Disadvantage of this EDM tool is that database is not protected from hacker.[4]

Vírseda Rafael del Vado (2010), In this paper author discussed an interactive visualization tool, which will be helpful for educator. The main benefit this tool is to helps computer science students to understand the conceptual knowledge about data structure algorithms. For development of this tool they used java.[5]

III Methods

In this we implement various algorithms of tree and graph data structure. Various algorithms that we implemented in tree they are Binary search tree (insertion, deletion, search), AVL (insertion, deletion, search), B-Tree (insertion, deletion, search), Heap (insertion, deletion, search), Red- Black Tree (insertion, deletion, search) and in graph they are BFS(Breath First Search), DFS (Depth First Search), minimum path algorithms i.e.

Dijkstra's Algorithm and Floyd – Warshall Algorithm and minimum cost spanning tree algorithms are Kruskal Algorithm and Prim's Algorithm. Now we discussed all one by one.

BST :- BST referred as binary search tree, it is also known as ordered binary tree. it is node based data structure and each node has not more than two child i.e. only left child, only right child and both left child and right child. the left sub tree element values is always less than root node value and right sub tree element value is always greater than root node value.

left tree elements < root element < right tree elements

AVL tree: AVL referred as Adelson – Velskii and Landis, it is also called balanced binary search tree. Two main properties of AVL tree are first height of each node is always lies between -1 to 1 means -1, 0 and 1. Second properties are balanced binary search tree means every time when node is inserted or deleted rotation is performed in tree if it is necessary. These rotations are one way rotation and two way rotation. In one way rotation we have LL rotation, RR rotation, LR rotation, RL rotation

Balance factor = height of left tree – height of right tree (values lies between -1 to 1)

B-Tree: it is another special type of data structure, in which one node have more than one values and further they are divided into more than two nodes. Suppose node have two values than they divided into three parts and the values of root node are x and y and child node have values lies between x and y means x1 is x<x1<y

Heap: Heap is a complete binary tree. Heap is two types' min heap and max heap. In min heap all parent node have less values as compared to its child nodes. In max heap all parent nodes have greater value as compared to its child node.

In heap nodes are represent as : Parent of x[i] = x[i/2]

Left child of x[i] = x[2i] and Right child of x[i] = x[2i + 1].

Red Black Tree: Red black tree is another special type of balanced binary search tree with nodes colored red or black. the basic properties of red black tree are every node have colored either red or black, every leaf (Null) is black, if node is red than its both child node color is black or vice versa and every simple path from a node to a successor leaf contain same number of black nodes.

BFS : BFS refer as Breadth first Search, it is traversing techniques of graph. The basic function of algorithm to traverse the graph as close as possible to the start node. BFS starts from top node. To implement BFS algorithm we used queue.

Basic steps performed in the BFS are as follows: 1) insert the start node in the queue, 2) Remove element from the queue and execute them, 3) inserted unvisited neighbors' node in the queue and 4) Repeat step 2 & 3 until queue is empty.

DFS: DFS refers as depth first search; it is another technique to traverse the graph. This algorithm starts from bottom. To implement this algorithm stack is used.

Basic steps of algorithm are: 1) Push the bottom element of the graph in the stack. 2) Pop the top element from the stack. 3) Push non visited node in the stack and repeat steps again until stack is empty.

Kruskal Algorithm: This algorithm is design by Joseph Kruskal. It is used to find minimum spanning tree in a graph. To determine the spanning tree in a graph, firstly arrange all edge according to the its weight in list (X) and start from smallest one, second select an edge from list (X) who is connected to the starting edge, add this edge in spanning tree list (T) and check it avoid the cycle. Last step is to repeat step second until all edges are covered.

Prim's algorithm: It is used to find minimum spanning tree in a graph. To find minimum spanning tree from graph firstly start from any random edge and add, second find the edge that have minimum weight rest available and add to tree list(T) and avoid cycle. Last step cover all edges and then stop.

Dijkastra algorithm: Djikstra's algorithm is created by Edsger Dijkastra. It is used to solve the problem of finding the shortest path from given point to all edges in graph. This algorithm can apply on directed and undirected graph but graph should be weighted and weight should be positive only. It is the solution to the single-source shortest path problem.

Floyd Warshall Algorithm : The Floyd-Warshall algorithm is designed by Stephen Warshall and implemented by Robert W. Floyd and P. Z. lngeraman. It is used to find shortest paths between all pairs of vertices in an edge weighted directed graph.

To developing these algorithms we used ASP.NET (.apsx) with java script and for backend database we used MS SQL server 2010.

IV Results and Discussion

This visualization tool has two parts: 1) visualization of algorithms 2) audio comments. In this visualization tool we visualized various algorithms and we discussed each algorithm with its results.

In BST (Binary Search Tree) we performed three operations i.e. insertion, deletion, search. In insertion operation we firstly insert the value "67" in insertion text box and then click on the insert button (step 1) and then algorithm find its location (step 2) and insert 67 that location (step 3) similarly deletion operation is performed. The screen shot of these steps as fallows in fig 1.

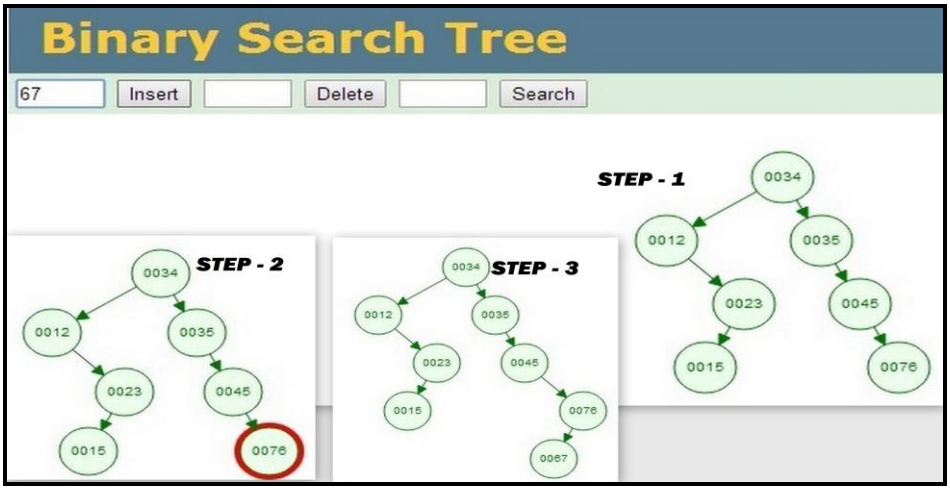


Figure – 1 Screenshot of binary search tree

Second we have AVL tree and in this we performed three operations i.e. insertion, deletion, search. In insertion operation user insert one number i.e. "92" and then click on the insert button (step 1) algorithm check its exact position and then insert on that location (step - 2) and then rearrange the node values and tree (step - 3), similarly we performed deletion. The screen shot of these steps as fallows.

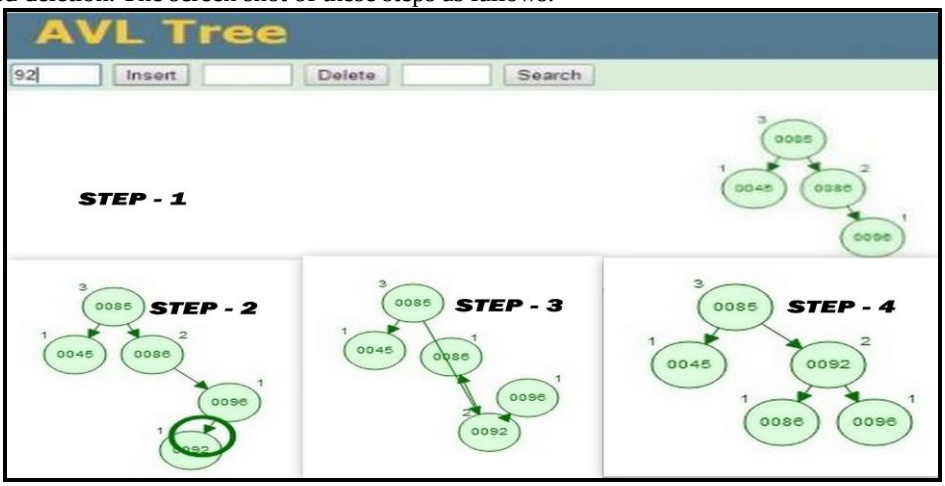


Figure – 2 Screenshot of AVL

Third we have B-tree we performed three operations i.e. insertion, deletion, search. In this implement 3 degree and 4 degree B-tree. In insertion module we insert value "74" and then click on then insert button algorithm (step - 1) then check element position (step -2) and insert value in node (step - 3) and then divided node and rearranged tree if needed. The screen shot of these steps as fallows.

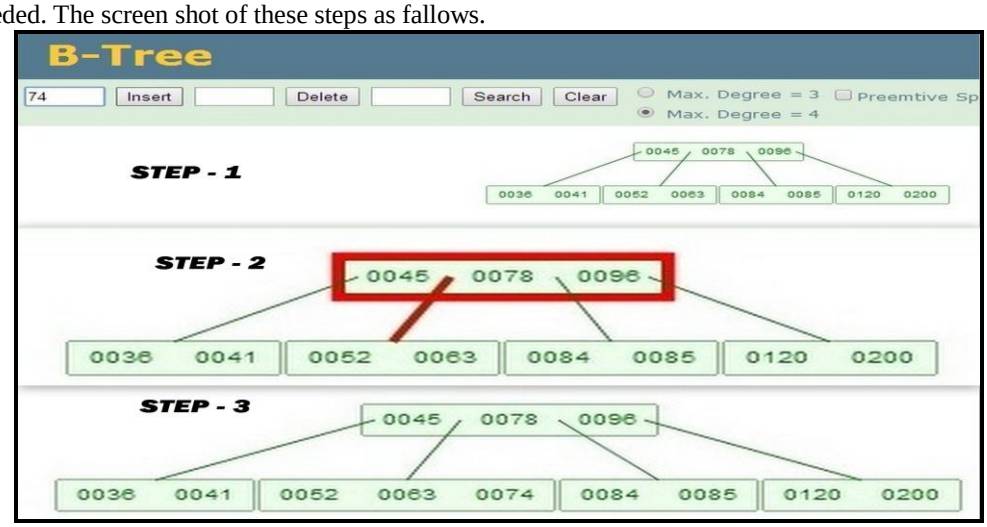


Figure – 3 Screenshot of B- tree

Next we have Red- Black tree and in this we performed three operations i.e. insertion, deletion, search. In insertion operation user insert value "54" in insertion text box (step -1) and the algorithm check its exact position

(step -2) and then insert on that position (step - 3) and then change node colour red to black or visa versa and then rotate tree if required. similarly deletion operation is performed.

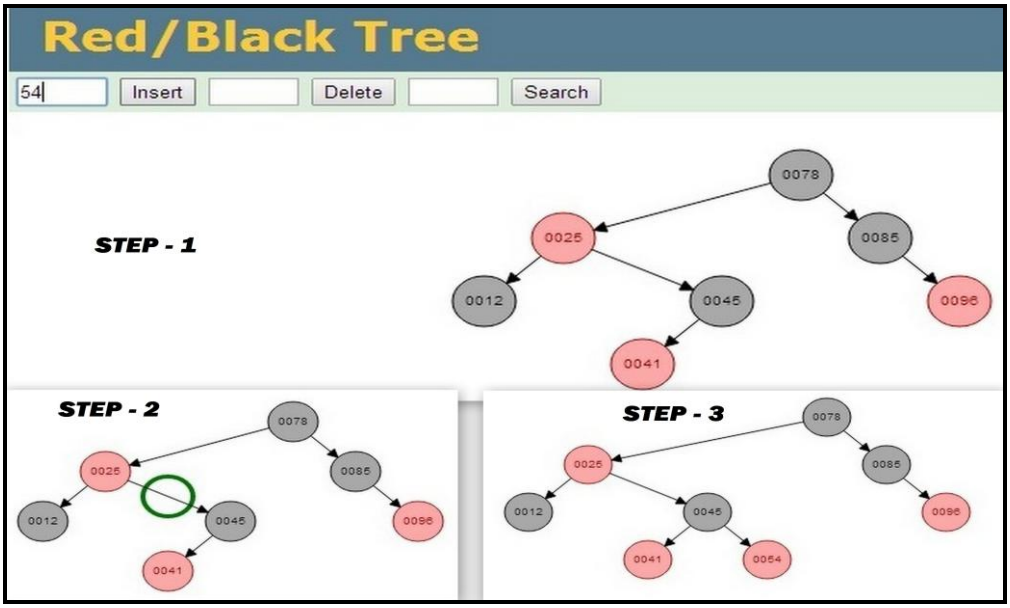
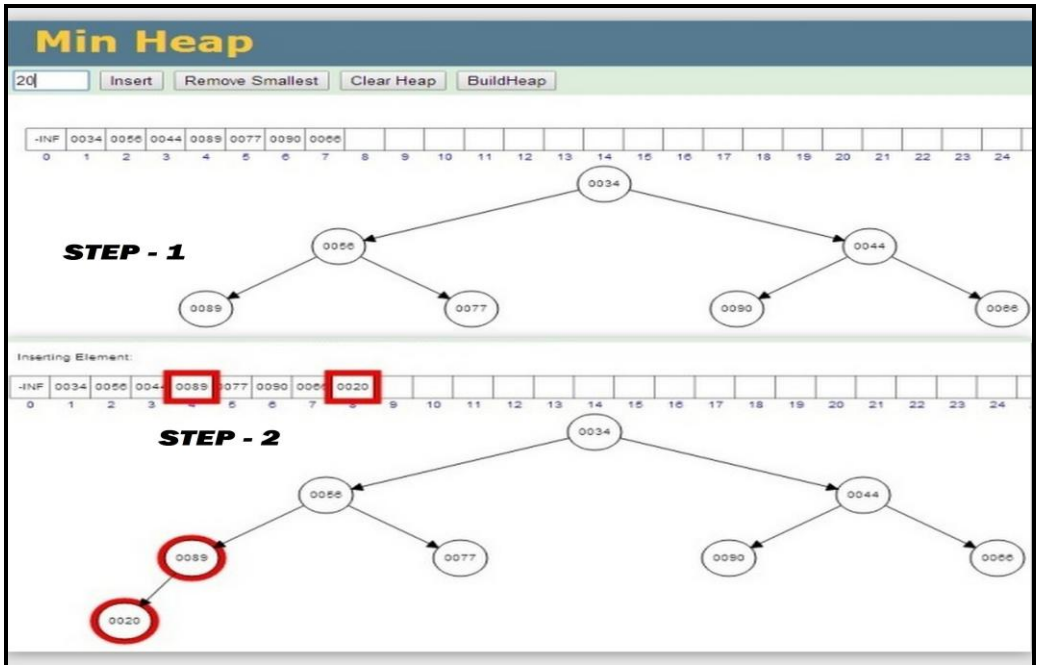


Figure – 4 Screenshot of Red Black tree

Next we Heap in this we implement three operations i.e. insertion, deletion, search. In insertion we insert value 20 and then click on insert button (step -1), next step -2 insertion is done at last node of tree considering property of complete binary tree, in step -3 rearrange the node values while considering every parent node smallest to its both child node and in last step -4 element "20" placed in its exact position. In case of deletion user simply click on the remove smallest n root element deleted and then rearrangement in tree executed automatically according to algorithm steps. The screen shot of these steps as fallows.



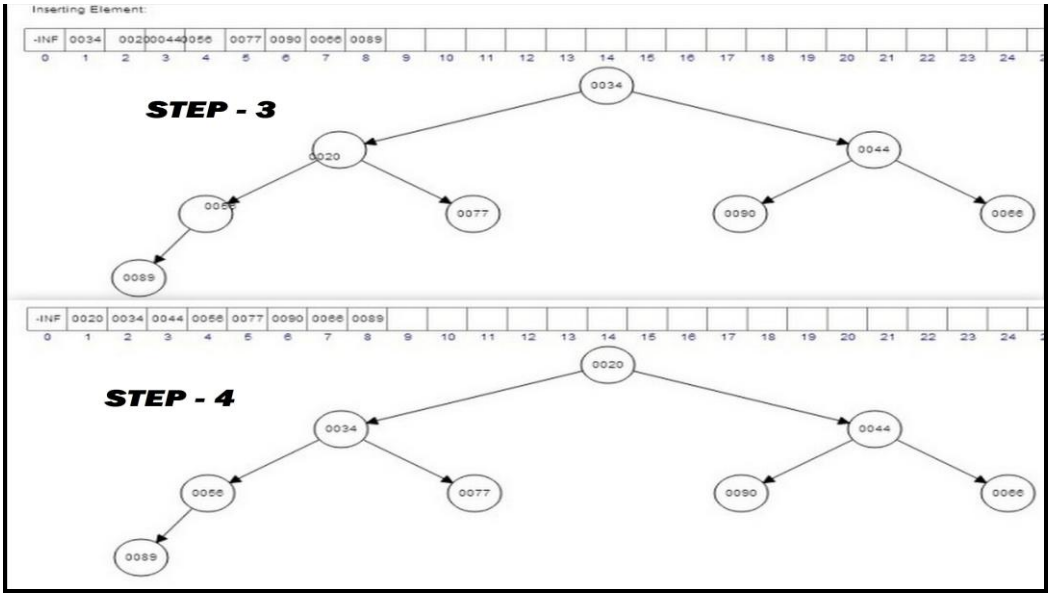


Figure – 5 Screenshot of Heap

Next we have Graph algorithms of traversal i.e. BFS (Breadth first Search) in this we apply visualization two types of graph i.e. directed and undirected graph. In this we represent graph in three forms logical representation, adjacency list representation and adjacency matrix representation and to reuse tool again we have new graph option. To start algorithm user entered starting node i.e. 1 and then click on the start BFS (step - 1) and then algorithm execution started. in next step – 2 inserting non visiting element in queue and then executing and in last step – 3 shows output in graph highlighted with blue line. The screen shot of these steps as fallows.

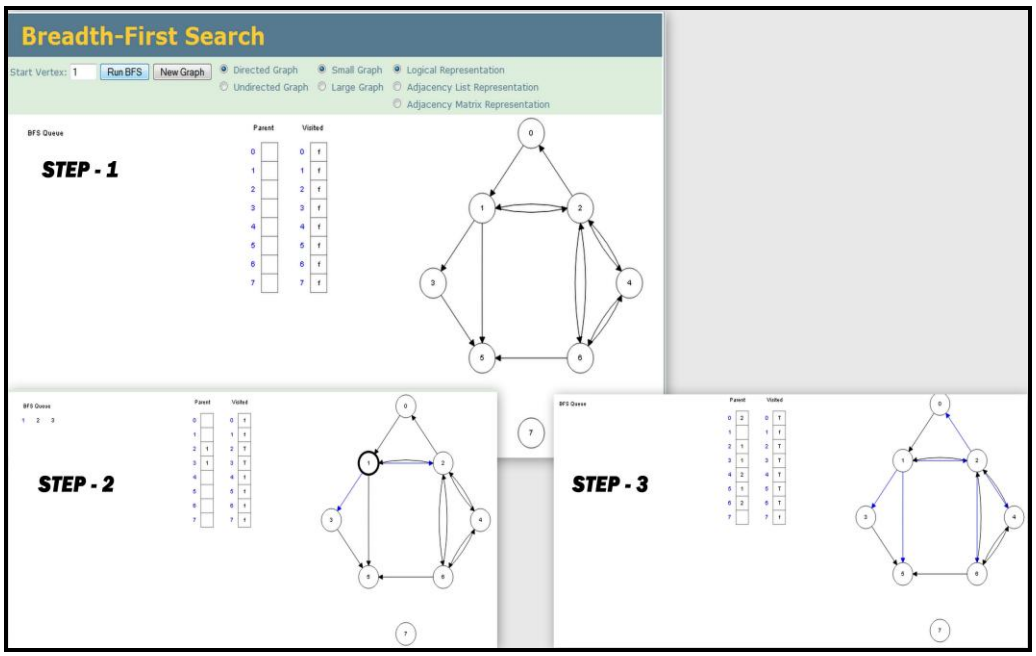


Figure – 6 Screenshot of Breadth – First Search

Next in Graph algorithms of traversal we have DFS (Depth first Search). In this we simply entered the node value "6" and then click on the run DFS button (step – 1) and then visualization started. in next step – 2 insert non visited nodes in the queue and execute it and in last step – 3 shows output of algorithm and path are highlighted by blue lines. The screen shot of these steps as fallows.

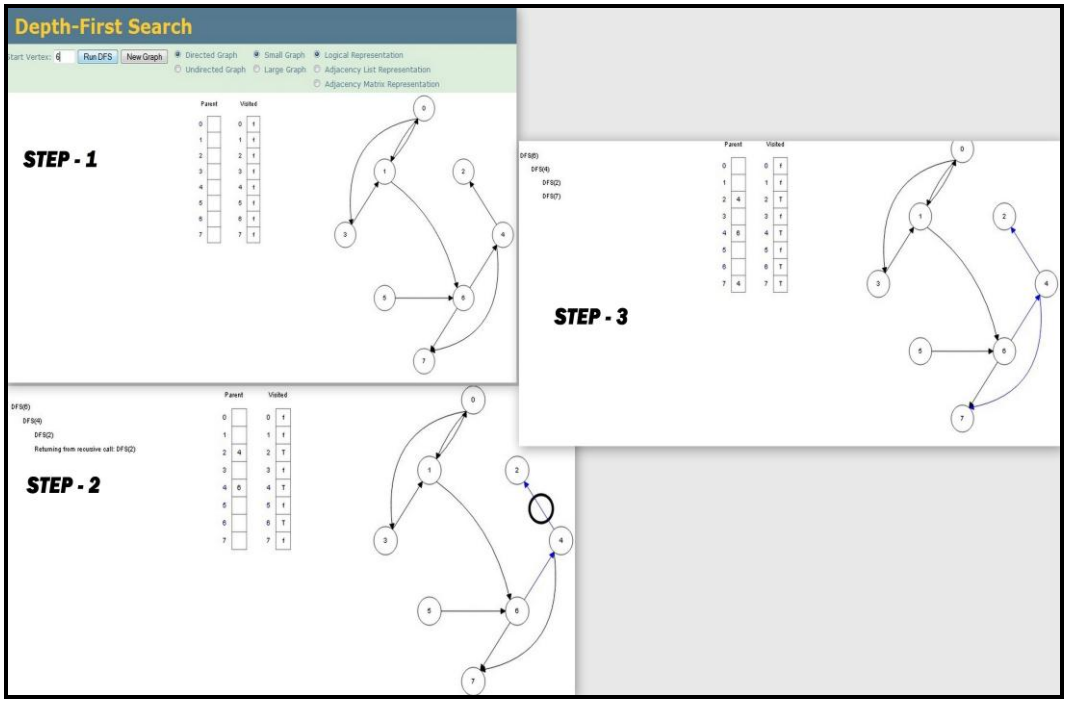


Figure – 7 Screenshot of Depth First Search

Next in Graph algorithms we have Kruskal algorithm that find minimum cost spanning tree. In this we simply click on the "Run Kruskal" button (step -1), then visualization started (step -2) and last step -3 output spanning tree is highlighted in red colour. The screen shot of these steps as fallows.

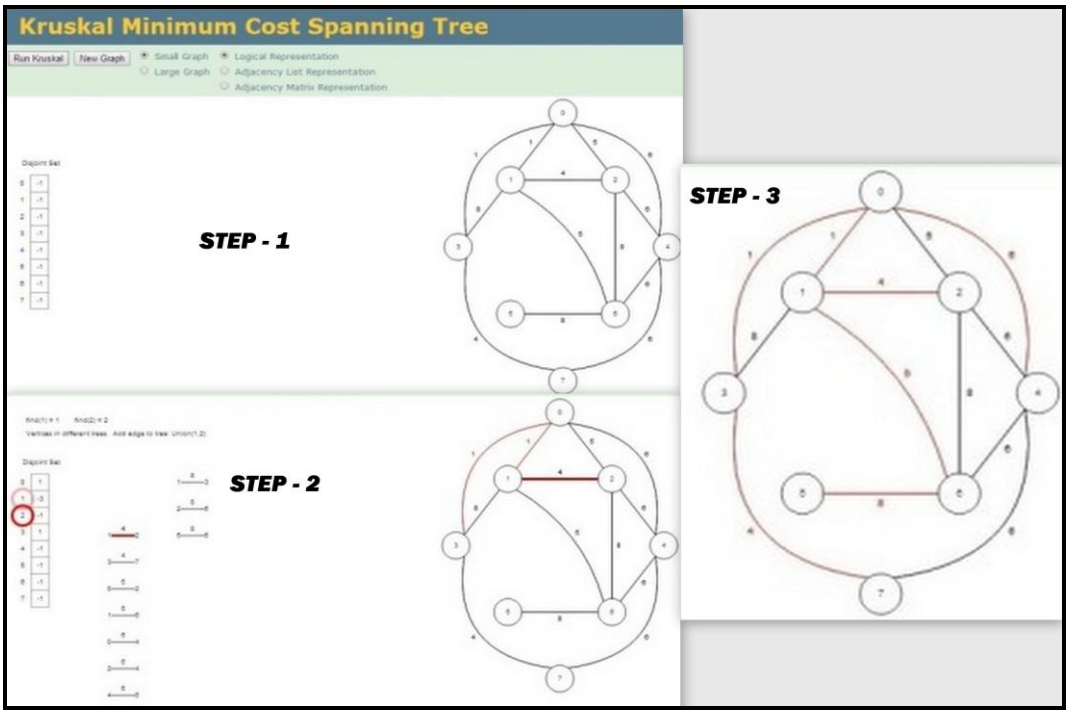


Figure – 8 Screenshot of Kruskal Algorithm for minimum spanning tree

Next in Graph algorithms we have Prim's algorithm that find minimum cost spanning tree. In this we insert staring node value "3" in text box then click on the "Run Prim's" (step -1), then visualization started (step -2) and last step – 3 output spanning tree is highlighted in red colour. The screen shot of these steps as fallows.

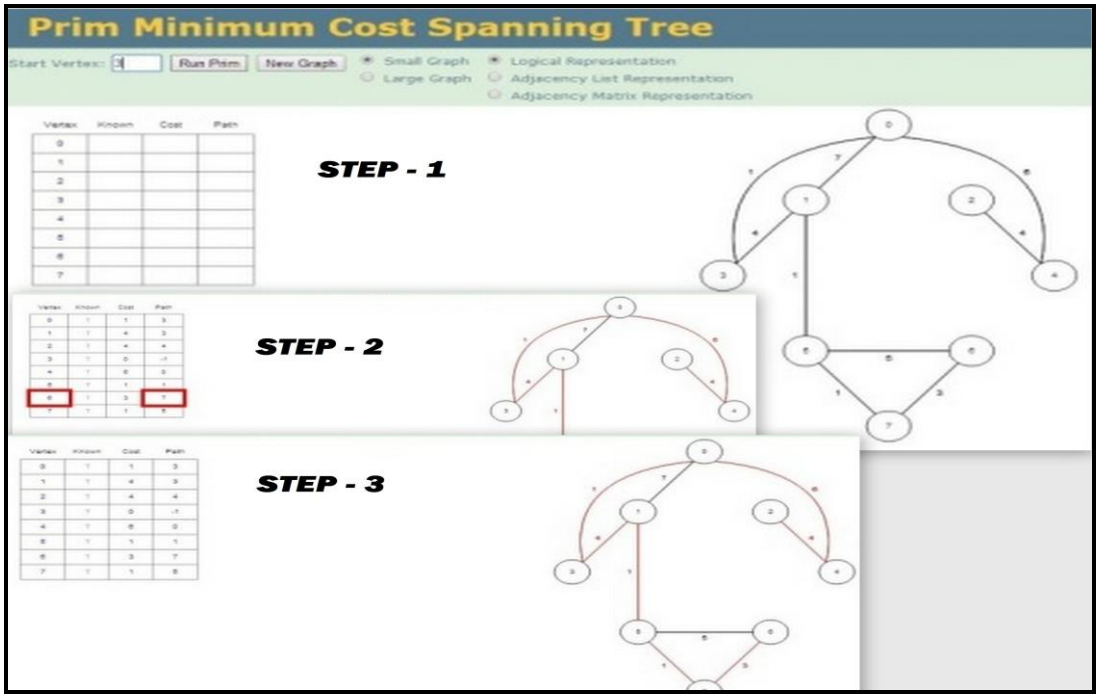


Figure – 9 Screenshot of Prim's Algorithm for minimum spanning tree

Next in Graph algorithms we have Dijkstra shortest path algorithm that find shortest path in the graph. In this we insert node value i.e. "4" in the text box then click on the "Run Dijkstra" button (step – 1), then visualization started(step – 2) and in step -3 output with path from 4 node to every node present in graph and if some node don't have path then output shows no path . The screen shot of these steps as fallows.

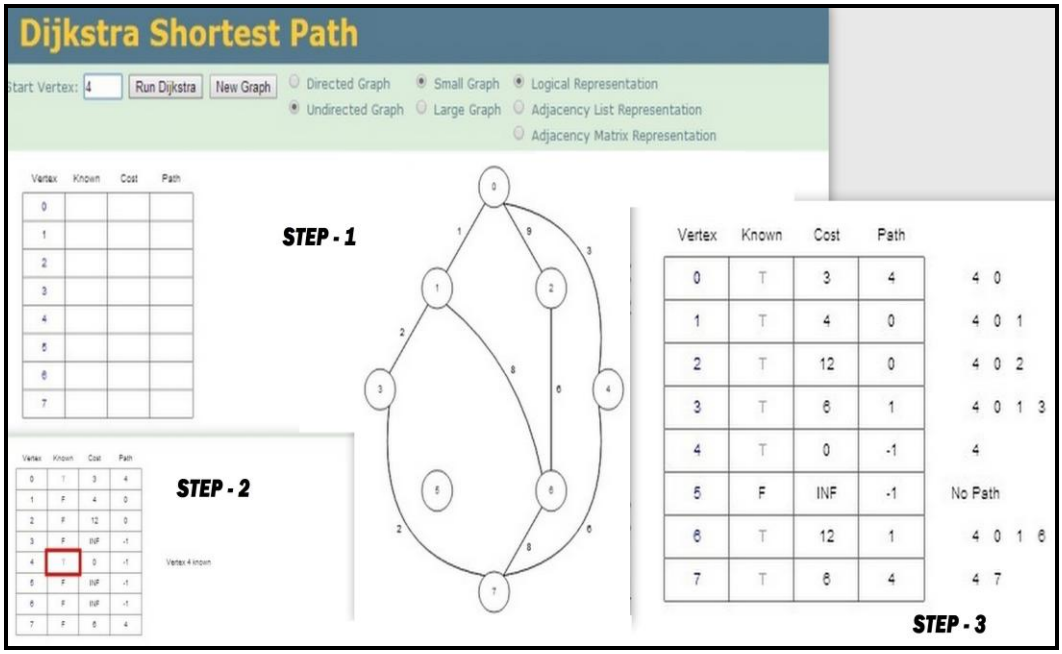


Figure – 10 Screenshot of Dijkstra algorithm for shortest path

Next in Graph algorithms we have Floyd- warshall algorithm that find all pairs shortest path in the graph. In this we simply click on the "Run Floyd- warshall" button (step - 1), then visualization started (step – 2) and step – 3 output will shows all nodes path weight age to all another's nodes in graph. The screen shot of these steps as fallows.

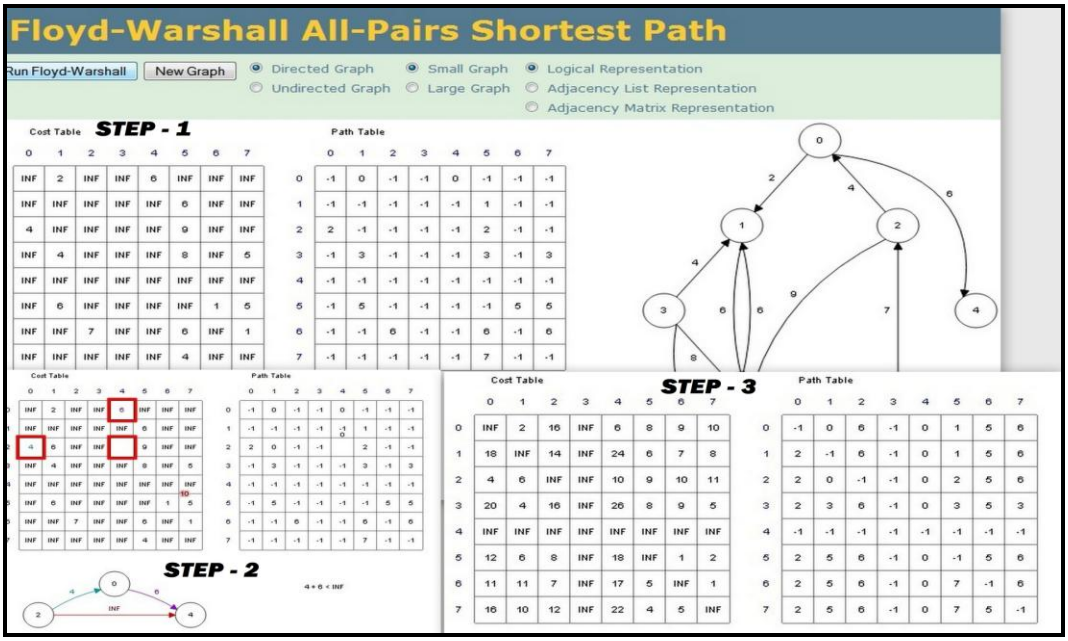


Figure – 11 Screenshot of Floyd - Warshall Algorithm for all pairs shortest path

In next we have feedback in which student give its feedback according to feedback form question and then submit feedback online and if student have some query then they submit their query online with its email id and educator revert back on his/her mail id. Main advantage of this tool is student can used this tool in campus or out campus due to online availability of tool. By using this tool they more easily clear the concept without help of any one and also improve its performance in exams and test.

V Conclusion and Future work

In this paper we discussed the visualization tool of tree and graph algorithms of data structure integrated with audio instruction. main algorithm implemented are BST(insertion, deletion, search), AVL(insertion, deletion, search), B-tree(insertion, deletion, search), Red- Black tree(insertion, deletion, search), Min heap(insertion, deletion, search), BFS, DFS, Prim's, Kruskal, Dijkstra and Floyd warshall. It is web based tool and student can do self study at home using this tool and if they have some query then they send online query to educator and they answered on his/her email id.

In future work we can add online testing module for the student and also design one platform where student draw its own data structure like AVL tree, Red- black tree and so on and tool indentified whether the draw data structure in correct or not.

References

[1] Meolic Robert (2013), *"Demonstration of Sorting Algorithms on Mobile Platforms"*, Proceedings of International Conference on Computer Supported Education, pp. 136-141.

[2] F. Alhosban (2012), "*Aural Instruction with Visualization in E-Learning*", Proceedings of Frontiers in Education Conference.

[3] P. Crescenzi, A. Malizia, M. Cecilia Verri, P. Díaz and I. Aedo (2012), "*Integrating Algorithm Visualization Video into a First-Year Algorithm and Data Structure Course*" , International Forum of Educational Technology & Society, pp: 155- 124.

[4] A. Traian, F. Adrian, G. Arpad (2011), "Web- Based Technologies for Online E-Learning Environment", Proceedings of 7th International scientific conference eLearning and Software for Education Bucharest.

[5] V. Rafael d. V. (2010), *"A Visualization Tool for Tutoring the Interactive Learning of Data Structures and Algorithmic Schemes"*, Proceedings of the 41st ACM technical symposium on Computer science education, pp: 187-191.