ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 4, Issue 4, pp: (57-67), Month: October - December 2016, Available at: www.researchpublish.com

Minimum Spanning Tree Algorithm

[1]D.Mishra, [2]S.P.Behera, [3]S. Bhattacharjee

Department of Mathematics C.V Raman (Autonomous) Engineering College, Bhubaneswar, Odisha
Ravenshaw University, Cuttack, Odisha

Abstract: **Given an edge-weighted graph, a spanning tree of the graph is a tree containing all the vertices and some or all the edges. The minimum spanning tree (MST) or minimum weight spanning tree (MWST) problem calls for finding the spanning tree whose weight is less than or equal to the weight of every other spanning tree. In this paper, we have presented an algorithm for finding minimum spanning tree which is different from the traditional Kruskal's and Prim's algorithm. In this algorithm we transform the given graph into a forest and then the minimum spanning tree is obtained from the forest. We have also implemented the proposed algorithm and the Kruskal's algorithm to find minimal spanning tree using Java programming language and the illustrations are demonstrated through a Java applet. We have presented some numerical examples to explain the solution procedure.**

Keywords: **Graph, Spanning Tree, Minimum Spanning Tree, Kruskal's algorithm.**

1. INTRODUCTION

Minimum spanning tree algorithm have direct applications in the design of networks including computer networks [1], transportation networks and it also use the area like cluster analysis [2], can provide an approximate solution to minimum spanning tree problem etc.

Kruskal's and Prims [3] are two well known algorithms to find the minimum spanning tree from a graph. Pettie and Ramachandran [4] proposed an optimal greedy algorithm to find minimal spanning tree. In this paper we present a new algorithm for solving minimum spanning tree problems which will work first finding forest formation and then it will convert forest into spanning tree.

2. PRELIMINARIES

In this Section we provide some basic definitions and concepts.

Graph**:** A graph G consists of a set V of vertices (nodes) and a set E of edges (arcs).We write G = (V, E), V is a finite and non-empty set of vertices. E is a set of pairs of vertices, their pairs are called edges.

Weighted graph**:** A graph G is said to be weighted if each edge e in G is assigned a non-negative numerical value w (e) called the weight or length of e.

Tree: A tree is a connected acyclic (i.e. free from cycles) graph.

Minimum Spanning tree**:** The Minimum spanning tree (MST) of a graph defines the cheapest subset of edges that keeps the graph in one connected component.

Array**:** An array can be defined as a finite collection of homogeneous (similar type) elements. The elements in an array are always stored in consecutive memory locations.

Stack**:** A stack is a non-primitive linear data structure. Here deletion and insertion of the element is done at one end, known as top of stack. Here the most frequently accessible element in the stack is the top most elements, whereas the last accessible elements are the bottom of the stack

3. AN IMPROVED MINIMUM SPANNING TREE ALGORITHM

In this algorithm for solving minimum spanning tree problems which will work first finding forest formation and then it will convert forest into spanning tree.

3.1 Algorithm Description:

Initially every node in the graph is considered individually and the shortest edge from that node is taken into consideration. After that we will traverse the graph to find out the minimum edges from each node in the graph. We encounter the output. If the best case is taken into account, the output of this step would be the minimum spanning tree itself. If, that is not the case, as we traverse from a start node to the last, we will encounter a break in the traversal. Thus all nodes will not be visited while traversal. And the outcome is the forest which is formed. Hence, to convert the forest so formed by the previous step of the algorithm, at the point of the break the next shortest edge is established. Then the same procedure from the traversal step is continued again till all the nodes are visited.

3.2 Algorithm:

The proposed algorithm to find minimum spanning tree is given below.

```
    Procedure Mspan(G)
    Input:
    /*G is the given weighted undirected graph
      N is the number of vertices
      u is the chosen Source vertex for each iteration.
      Y is an element in the array.
      a[ ] is the  array.
      v is the destination vertex.
    */
Output: Minimum spanning tree of the graph G
    Begin
For u =1 to N
     Visited [u] =0
    Choose e=min_{v∈G}(w(u,v))
     include e in the forest
 End for
   u=1; i=1;count=1;
   Visited[u] =1;
   Push u into Stack;
   Traverse(u);
   while  Count  < N  do
    if  Stack !=Empty
          a[]=Pop();
           i++;
          u=Stack[TOP];
         Traverse(u);
     End if
      if  Stack = Empty;
          Choose<Y,V'>
        /*<Y,V'> is the next   shortest path */
          Visited [V']=1;
```

```
            Count ++;
           Traverse(V');
        End if
      End while
Return
Traverse(u)
for  each non visited adjacent  vertex V;
        Visited[V]=1;
        Count ++;
        Traverse (V);
End for
      End.
```

3.3 An Example:

A TV cable company is in the processing of planning a network for providing cable TV service to five new housing development areas. The cable system network is summarized in figure-2. The numbers associated with each branch represent the miles of cable needed to connect any two locations. Node u_0 represents the cable TV relay station and the remaining nodes (u_1 to u_5) represent the five development areas. A missing branch between two locations implies, that is prohibitively expensive or physically impossible to connect the associated development areas. It is required to determine the links that will result in the use of minimum cable miles while guaranteeing that all areas are connected (directly or indirectly) to the cable TV station.

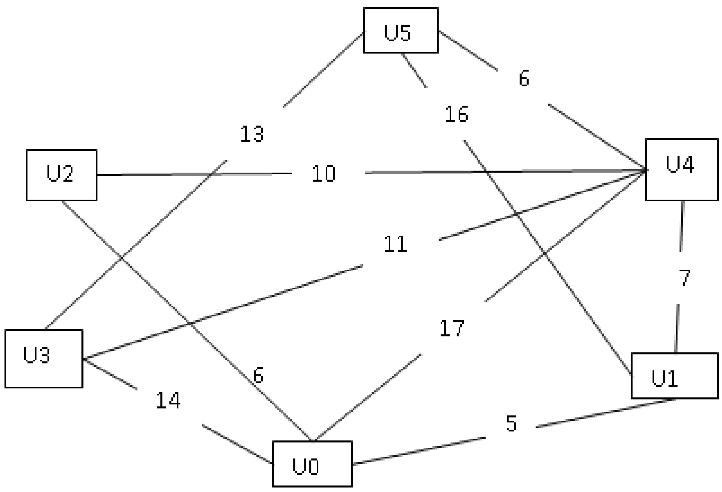
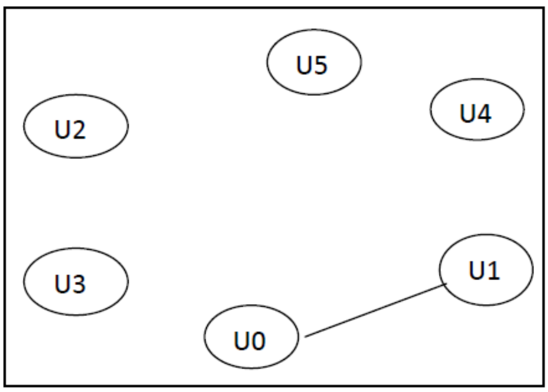
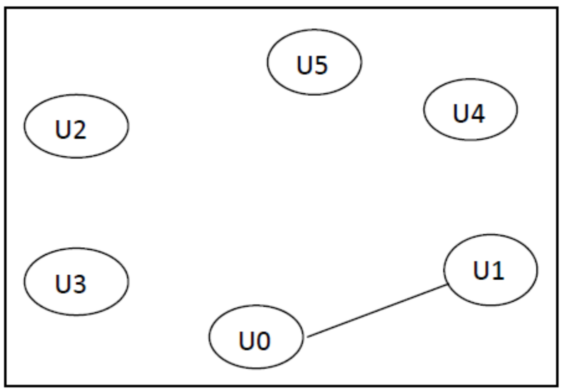


Figure-2: Example

STEP-1

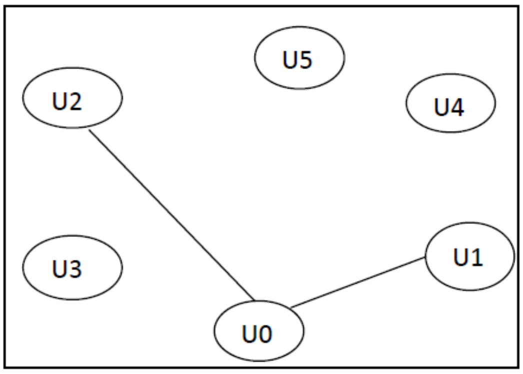


STEP-2

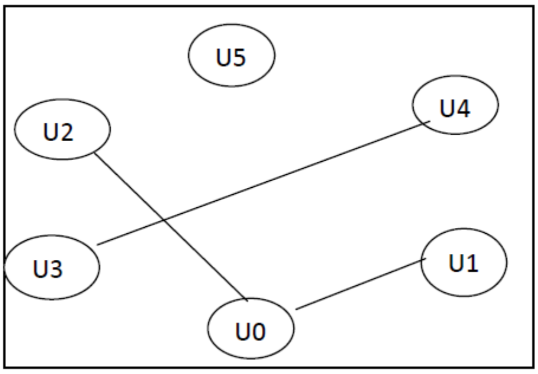


ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 4, Issue 4, pp: (57-67), Month: October - December 2016, Available at: www.researchpublish.com

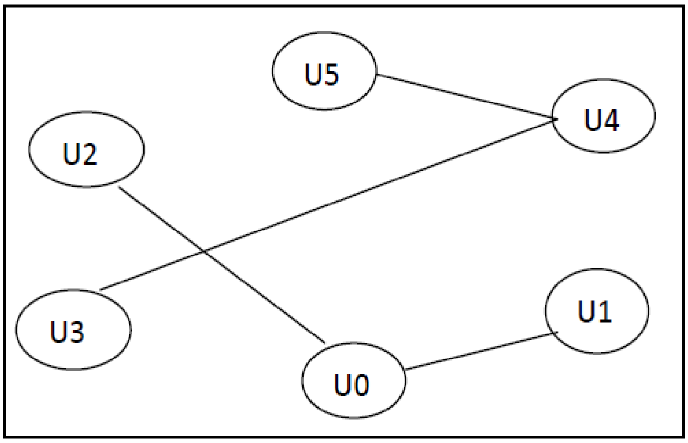
STEP-3



STEP-4

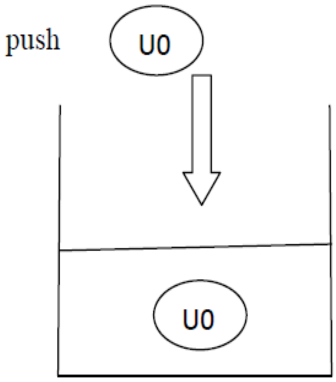


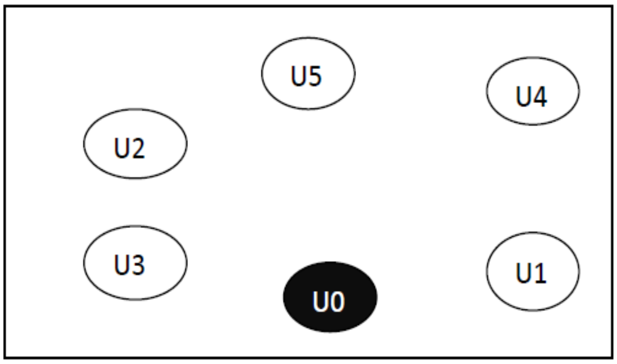
STEP-5



STEP-6

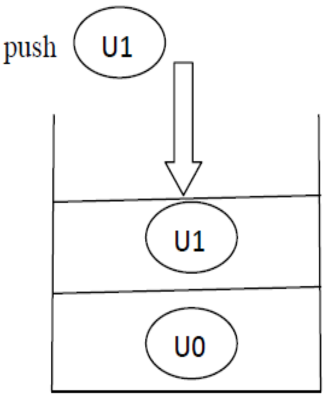
STACK

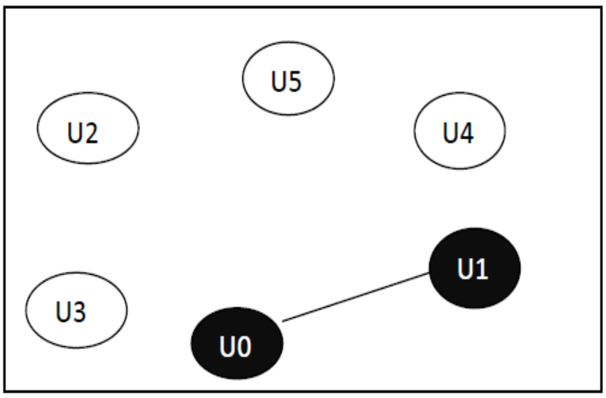




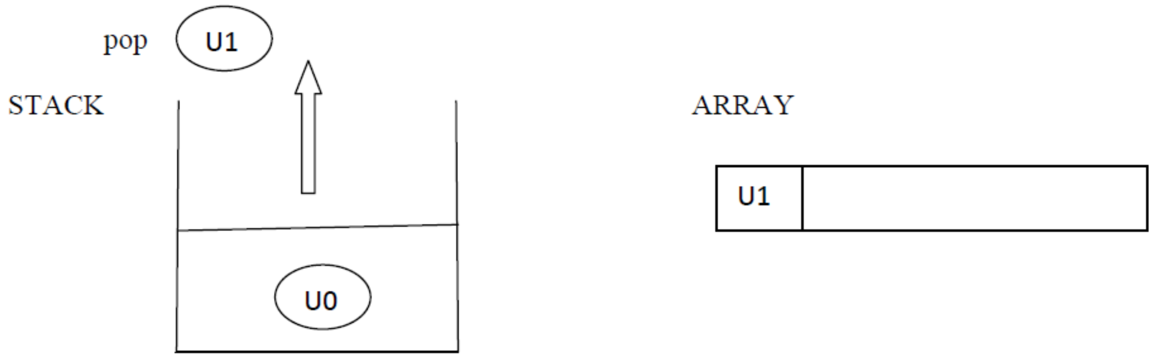
STEP-7

STACK

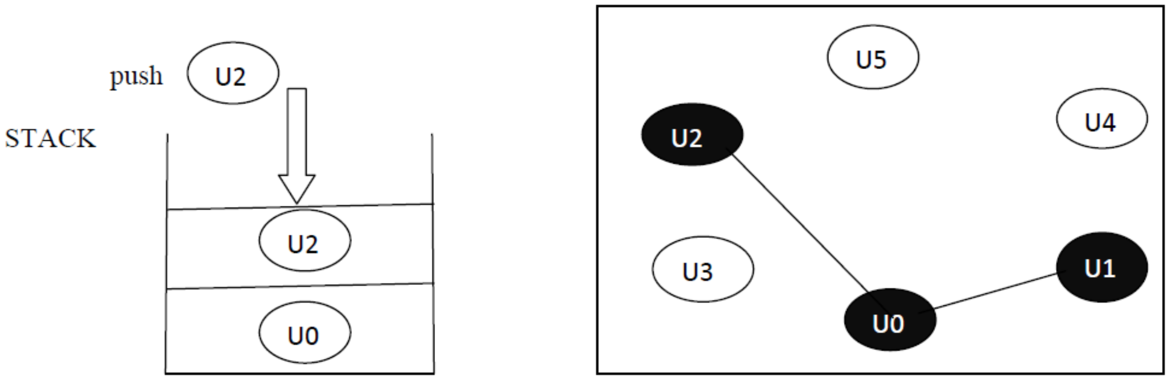




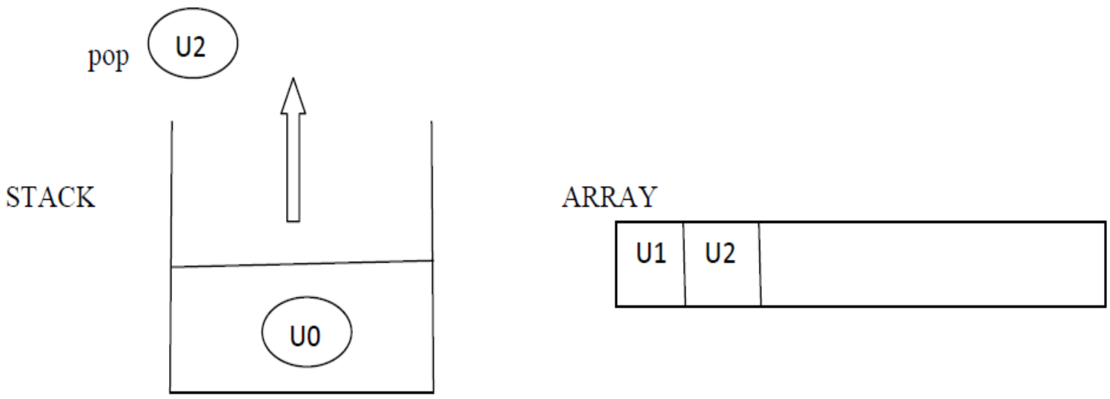
STEP-8



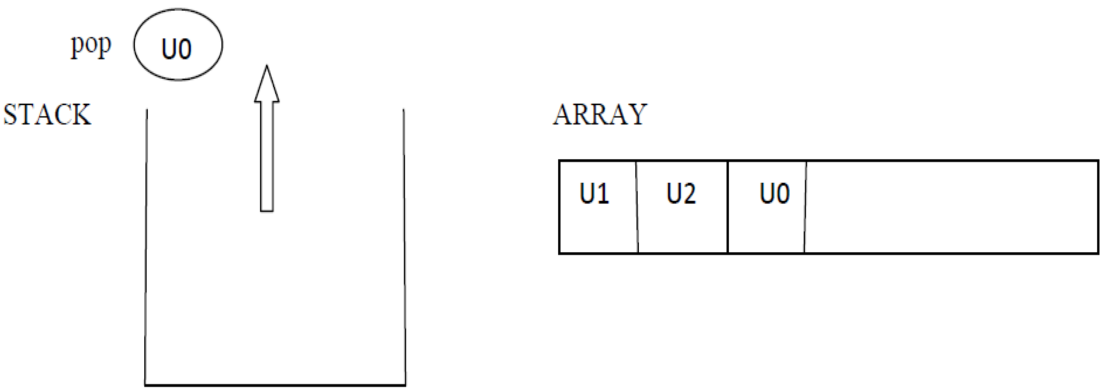
STEP-9



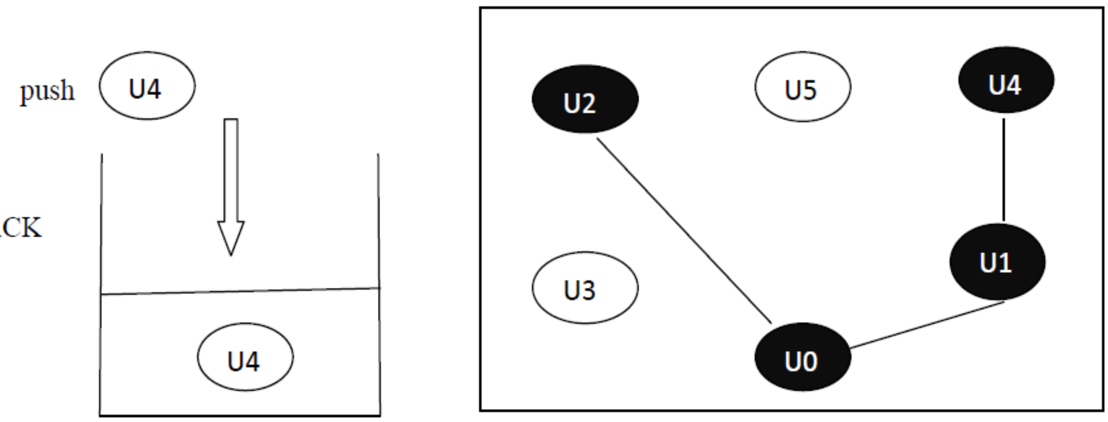
STEP-10



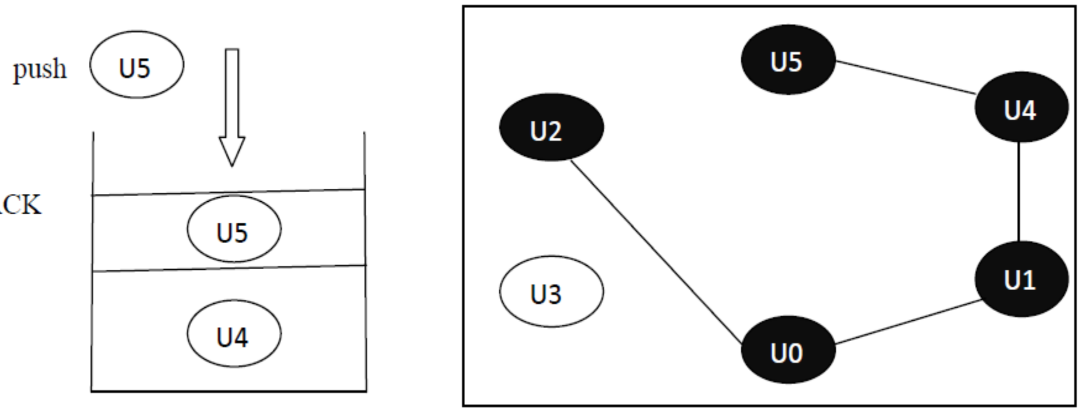
STEP-11



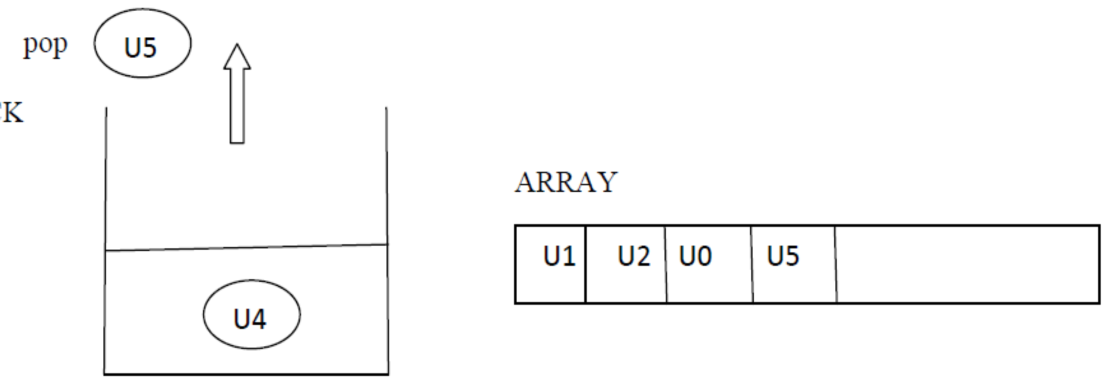
STEP-12



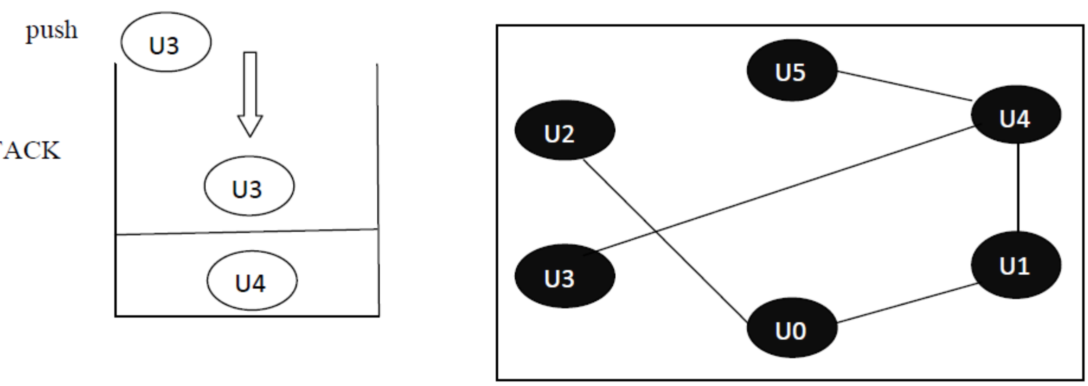
STEP-13



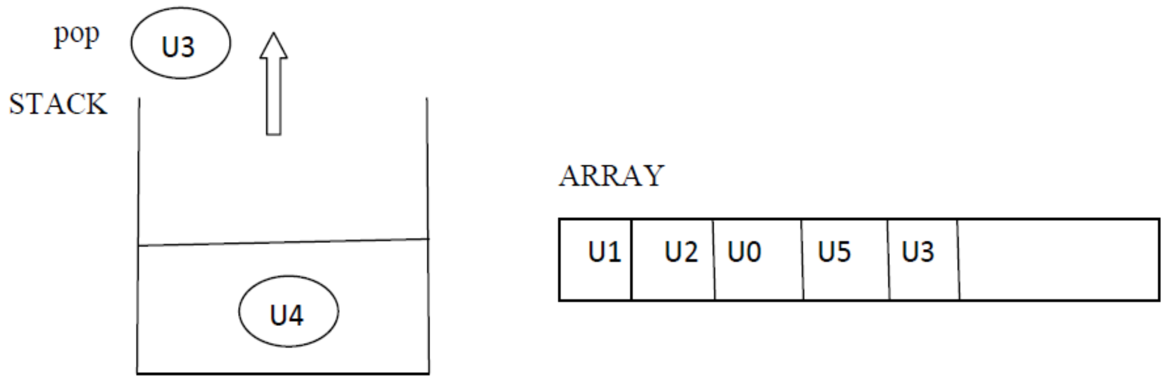
STEP-14



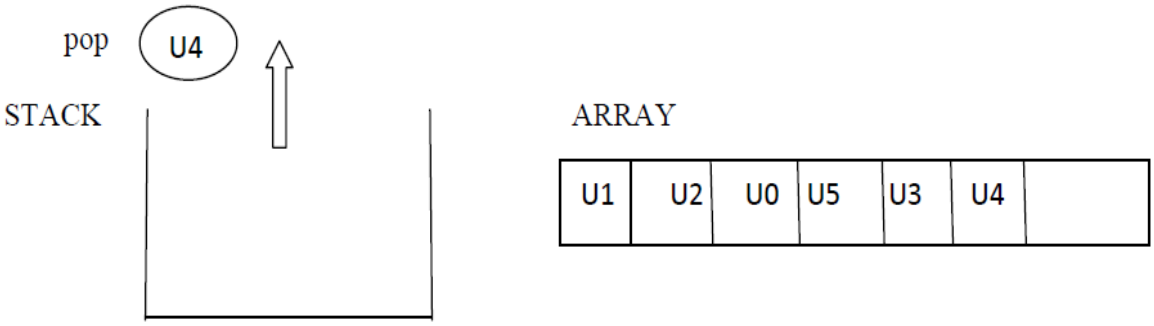
STEP-15



STEP-16

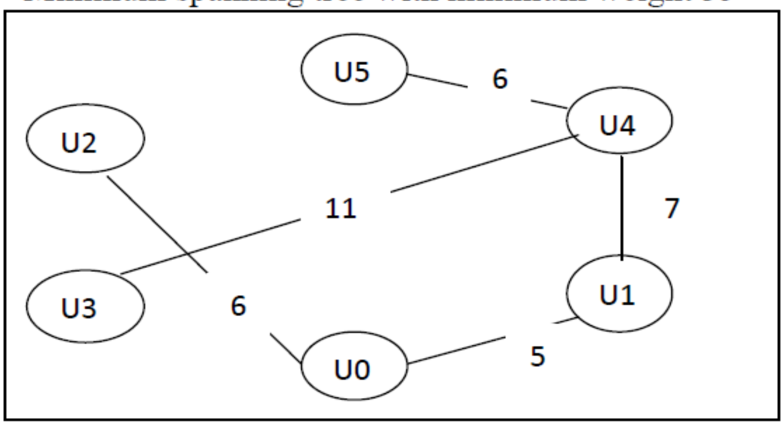


STEP-17



Final output:

Minimum spanning tree with minimum weight 35



3.4 COMPLEXITY ANALYSIS:

The first for loop in the algorithm is $|V|$. The traversing cost in the graph is $O(|E|\log|V|)$. Hence the total time complexity for the MSpan algorithm is $O(|E|^2 \log|V|)$, which is same as the time complexity of Kruskal's and Prim's [3] [5] algorithm.

4. EXPRIMENTS USING JAVA APPLET PROGRAMMING

Here we have designed JAVA APPLET programming based on Kruskal's algorithm and Mspan's algorithm compare with our result which gives us equal values as calculated example 3.3.The initial graph is shown in figure 3 and figure 7. The final result is shown in figure 5 and figure 8.

4.1 Implementation of Kruskal's Algorithm:

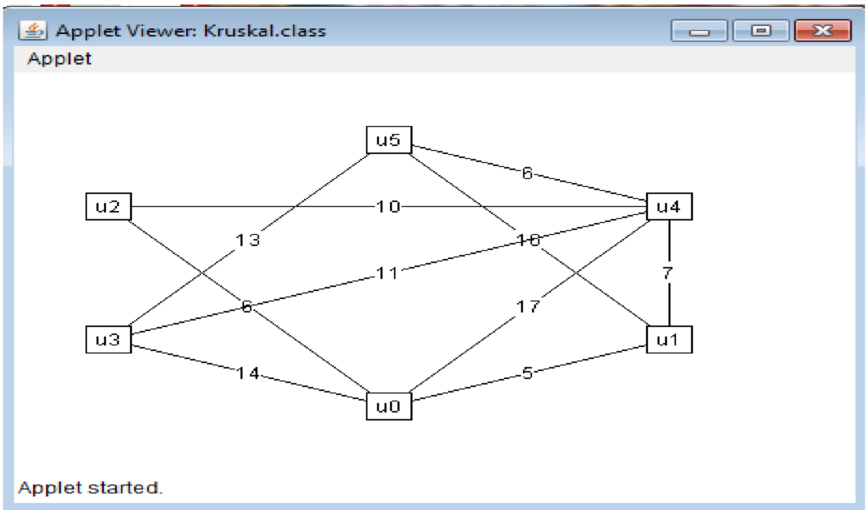


Fig 3: Initial Step with reference to Example-3.3

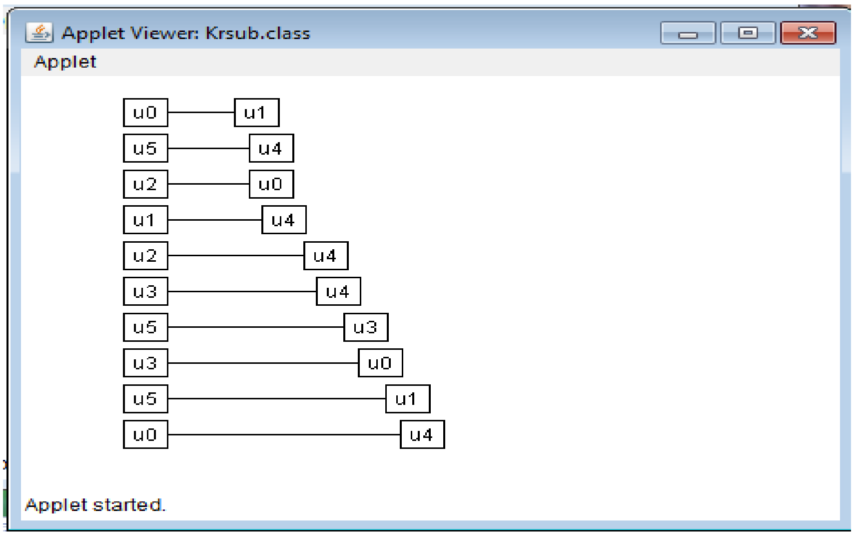


Fig 4: Initial Step with edge sorting as in Fig 3

Final output

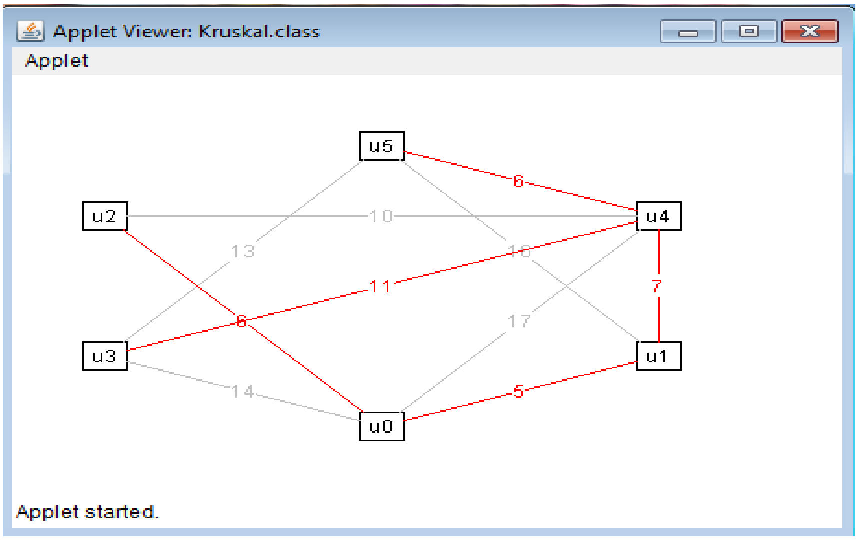


Fig 5: minimum spanning tree for example 3.3

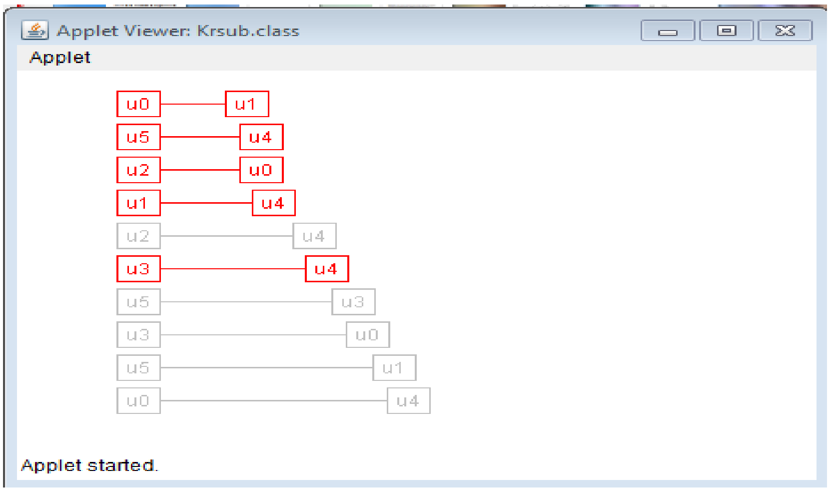


Fig 6: Optimal edge associated with minimum spanning tree as in figure 5

Here the weight of Minimum Spanning Tree =5+6+6+7+11 =35

4.2 Implementation of Mspan Algorithm:

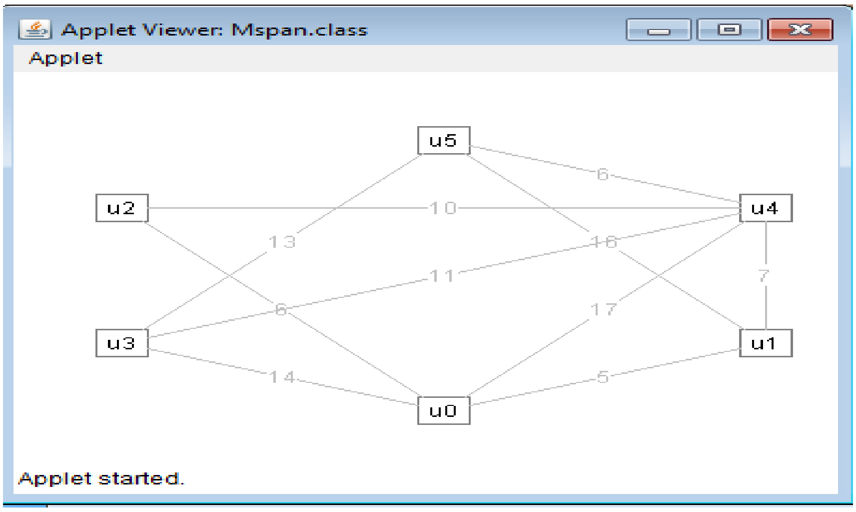


Fig 7: Initial Step with reference to Example-3.3

Final output

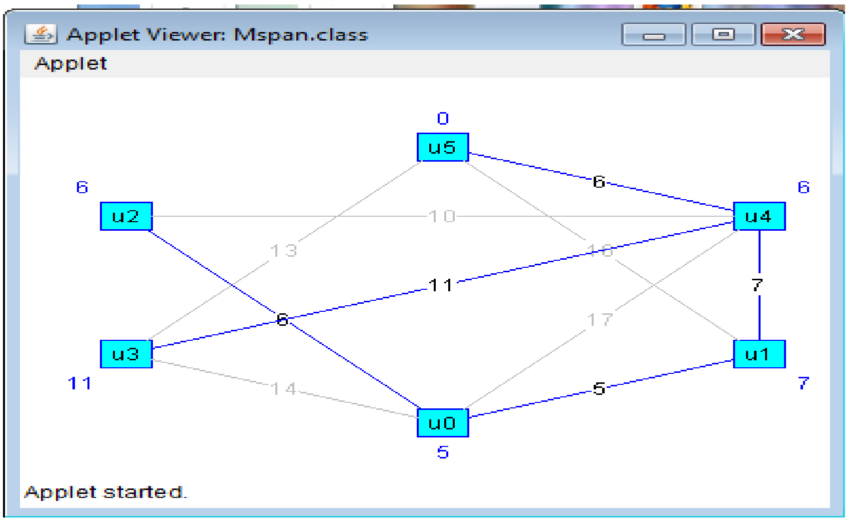


Fig 8: Minimum spanning tree obtained by Mspan

Here the weight of Minimum Spanning Tree =5+6+6+7+11 =35.

5. CONCLUSION AND FUTURE SCOPE

5.1 Conclusion:

In this paper we have developed an algorithm which is named as “MSpan” to find out the minimum spanning tree from a weighted graph. The time complexity of MSpan is same as that of Kruskal’s and Prims algorithm. The additional feature of MSpan algorithm is that, there no formation of loop. Hence running a check for occurrence of closed loop is completely avoided.

5.2 Future Scope:

In the future, we will explore and test our developed algorithm “Mspan” in various domains. It is an important combinatorial optimization problem which is improved dramatically in the last decade. The availability of reliable software, extremely fast and inexpensive hardware and high –level languages that make the modeling of complex problems faster have led to much greater demand for optimization tools. Keeping the above points of view our future work will more emphasize much larger problems on personal computers, much of the necessary data is routinely collected and tools exist to speed up both the modeling and the post optimality analysis.

REFERENCES

[1] Macdonald, P.J.; Almaas,E. and Barab´asi,A.-L.; Minimum spanning trees of weighted scale-free networks *Europhys. Lett*, 72 (2), pp. 308–314 (2005).

[2] Jana, P.K.; Naik, A., "An efficient minimum spanning tree based clustering algorithm," *Methods and Models in Computer Science, 2009. ICM2CS 2009. Proceeding of International Conference on* , vol., no., pp.1,5, 14-15 Dec. 2009,doi: 10.1109/ICM2CS.2009.5397966

[3] Thomas.H.Corman, Charles.E.Leisserson, Ronald.L.Rivest & Clifford Stein, “Introduction to Algorithms” Second Edition.

[4] Seth Pettie and Vijaya Ramachandran. 2002. An optimal minimum spanning tree algorithm. *J. ACM* 49, 1 (January 2002), 16-34. DOI=10.1145/505241.505243

[5] Feixue Huang; Pengfei Gao; Yu Wang, "Comparison of Prim and Kruskal on Shanghai and Shenzhen 300 Index Hierarchical Structure Tree," *Web Information Systems and Mining, 2009. WISM 2009. International Conference on*, vol., no., pp.237, 241, 7-8 Nov. 2009, doi: 10.1109/WISM.2009.56.

[6] Berge C. and Ghouila-Houri A., Programming, Games, and Transportation Networks. John Wiley, New York, 1965.

[7] Boruvka O.,and jistem O., problemu minimalm. Prace Moravske Prrodovedecke Spolecnosti, 3:37-58, 1926. In Czech.

[8] Boruvka.O Prspevek k resenotazky ekonomicke stavby elektrovodnch st. Elektrotechnicky obzor, 15:153-154, 1926.

[9] Chazelle B., A minimum spanning tree algorithm with inverse-Ackermann type complexity. J. ACM, 47(6):1028-1047, 2000.

[10] ChoquetG., Etude G., de certains r'eseaux de routes. Comptes Rendus Acad. Sci., 206:310-313, 1938.

[11] Cormen T.H, Leiserson C.E. , Rivest R.L., and Stein C., Introduction to Algorithms. MIT Press, 2001.

[12] Czekanowski J., Zur dierentialdiagnose der neandertalgruppe, korrespondez-blatt der deutschen gesellschaft fur anthropologie. Ethn. U Urg., 40:44-47, 1909.

[13] Dijkstra E.W., A note on two problems in connexion with graphs. Nu- merische Mathematik, 1:269-271, 1959.

[14] Dijkstra E.W., Some theorems on spanning subtrees of a graph. Indag. Math., XXII(2):196-199, 1960.

[15] Edmonds J, Matroids and the greedy algorithm. Math. Programming, 1:127-136, 1971.

[16] Florek K., Lukasziewicz L. , Perkal J., Steinhaus H. , and Zubrzycki S., Sur la liaison et la division des points d'un ensemble ni. Colloq. Math., 2:282-285, 1951.

[17] Graham R.L. and Hell P., On the history of the minimum spanning tree problem. Ann. Hist. Comput., 7(1):43-57, 1985.

[18] Jarnk V, jistem O., problemu minimalnm. Praca Moravske Prrodovedecke Spolecnosti, 6:57-63, 1930. In Czech.

[19] Kalaba R., On some communication network problems. In Proc. Symp. Applied mathematics, pages 261-280, 1960.

[20] Kotzig A. Suvisl'e grafy s minimalnou hodnotou v konecnom suvislom grafe. Casopis Pest. Mat., 86:1-6, 1961.

[21] Kruskal J.B,On the shortest spanning subtree of a graph and the traveling salesman problem. In Proc. Amer. Math. Soc., volume 7, pages 48-50, 1956.

[22] Lawler E., Combinatorial Optimization: Networks and Matroids. Holt, Rinehart & Winston, New York, 1976.

[23] Loberman H. and Weinberger A., Formal procedures for connecting terminals with a minimum total wire length. J. ACM, 4(4):428-437, 1957.

[24] Guan Mei-Gu. The method of eliminating cycles for fnding minimum spanning trees (in chinese). Shuxue de Shihian yu Renshi, 4, 1975.

[25] Nesetril J. A few remarks on the history of MST problem. Archivum Mathematicum, 33:15-22, 1997.

[26] Prim R.C, Shortest connection networks and some generalizations. Bell Systems Technical Journal, pages 1389-1401, 1957.

[27] Schmidt J.P, Siegel A, and Srinivasan A. Cherno-Hoding bounds for applications with limited independence. SIAM J. Discr. Math., 8(2):223-250, 1995.

[28] Sollin G, Problem de l'arbre minimum (unpublished). 1961.

[29] Rosenstiehl P., L'arbre minimum d'un graphe. In Theory of Graphs (Intl. Symp., Rome, 1966), pages 357-368. Gordon and Breach, New York, 1967.

[30] Kowalik L. and Kurowski M. , Short path queries in planar graphs in constant time. In Proc. 35th ACM Symp. on the Theory of Computing, 2003.

[31] Meyer U. and Sanders P. , Parallel shortest path for arbitrary graphs. In Proc. 6th Int'l Euro-Par Conference, LNCS 1900, pages 461-470, 2000.

[32] Meyer U. and Zeh N., I/O-efficient undirected shortest paths. In Proc. 11th Ann. Euro. Symposium on Algorithms (ESA), 2003.

[33] Thorup M., Quick k-median, k-center, and facility location for sparse graphs. In Proceedings Int'l Colloq. on Automata, Languages, and Programming (ICALP), LNCS 2076, pages 249-260, 2001.

[34] Cüneyt F. Bazlamaçcı, Khalil S. Hindi "Minimum-weight spanning tree algorithms A survey and empirical study" Computers & Operations Research, Volume 28, Issue 8, July 2001, Pages 767-785.

[35] Zahra Naji-Azimi, Majid Salari, Bruce Golden, Raghavan. S, Paolo Toth "Variable neighborhood search for the cost constrained minimum label spanning tree and label constrained minimum spanning tree problems" *Computers & Operations Research, Volume 37, Issue 11, November 2010, Pages 1952-1964.*

[36] Roberto Cordone, Gianluca Passeri "Solving the Quadratic Minimum Spanning Tree Problem *Applied Mathematics and Computation, Volume 218, Issue 23, 1 August 2012, Pages 11597-11612.*

[37] Francis J. Vasko, Robert S. Barbieri, Brian Q. Rieksts, Kenneth L. Reitmeyer, Kenneth L. Stott Jr. " The cable trench problem: combining the shortest path and minimum spanning tree problems "*Computers & Operations Research, Volume 29, Issue 5, April 2002, Pages 441-458*