# VGTool: Web Tool for Visualizing and Determining the Class of Gracefully Labeled Tree

Jannatul Maowa
Department of Computer Science
American International University-Bangladesh
Dhaka, Bangladesh

Sharifa Rania Mahmud
Department of Computer Science and Engineering
Military Institute of Science and Technology
Dhaka, Bangladesh

## ABSTRACT
A tree is a connected acyclic graph on *n* vertices and *n-1* edges. Graceful labeling of a tree is a labeling of its vertices with the numbers from *0* to *n-1*, so that no two vertices share a label, labels of edges, being absolute difference of the labels of its end points, are also distinct. There is a famous conjecture named Graceful tree conjecture or Ringel-Kotzig Conjecture that says "All trees are graceful". Almost 50-year old conjecture is yet to be proved. However, researchers have been able to prove that many classes of trees are graceful. In this paper, we have introduced a new web tool named VGTool which help many researcher to know which classes of trees already been proved to be graceful. Moreover, researcher can generate random tree by using this web tool and verify in which class this tree belongs. If it is belongs to some known class then web tool generate graceful labeling of this tree otherwise researcher can try to classify this tree. We hope this web tool will help researcher in a very useful way.

## Keywords
graceful labeling, graph theory, graceful tree conjecture, web tool

## 1. INTRODUCTION
In 1963, Ringel conjectured that Every tree with $m+1$ vertices decomposes $K_{2m+1}$ which has been known as Ringel's Conjecture. This problem is to date still unsolved. In 1967, According to Rosa, Kotzig conjectured a stronger statement than Ringel's, which has been known as Kotzig's Conjecture. Kotzig's Conjectured that, Every tree with m+1 vertices cyclically decomposes $K_{2m+1}$. In 1967, a paper was published by Rosa. Intention of Rosa behind publishing the paper [19] was for providing insight into Ringel's Conjecture for efficiently addressing the issue. The idea was to use a labeling of the vertices of a graph $H$ of order $m$ to show that it can cyclically decompose $K_{2m+1}$. Rosa referred this labeling as a valuation of the graph. Consider the following conditions where $O_G$ be a labeling of the vertices of $G$, $V_{OG}$ be the numbers assigned to the vertices of $G$ and $E_{OG}$ be the set of numbers assigned to the edges of $G$.

1) $V_{OG} \subseteq \{1, 2, \dots, n\}$,
2) $V_{OG} \subseteq \{1, 2, \dots, 2n\}$,
3) $E_{OG} \subseteq \{1, 2, \dots, n-1\}$,
4) $E_{OG} \subseteq \{x_1, x_2, \dots, x_n\}$, where $x_i=i$ or $x_i=2n+1-i$,
5) There exists $x \in \{1,2, \dots, n\}$, such that for an arbitrary edge $v_iv_j$ of the graph either $a_i \leq x < a_j$ or $a_j \leq x < a_i$, where $a_k$ is the label assigned to $v_k$ for each $k$.

From above conditions Rosa defines four types of labeling which are given below :

- A $\rho$-valuation satisfies conditions (2) and (4).
- A $\sigma$-valuation satisfies conditions (2) and (3).
- A $\beta$-valuation satisfies conditions (1) and (3).
- A $\alpha$-valuation satisfies conditions (1), (3) , and (5).

Graceful labeling is the other name of Rosa's $\beta$-valuation. In 1972, The term "Graceful Labeling" was first introduced by Golomb. A lot of work have been done by many researchers to prove the Graceful Tree conjecture (GTC). But The problem still remains open.

**Graceful Labeling of Tree:** A $n$ order tree is called graceful if there exists a labeling of its vertices's with the numbers from $0$ to $n-1$ such that the set of absolute values of the differences of the numbers assigned to the ends of each edge is the set $\{1, 2, \dots, n-1\}$.

An example of graceful labeling of tree with $n=7$ vertex is illustrated in the figure (see Figure 1).

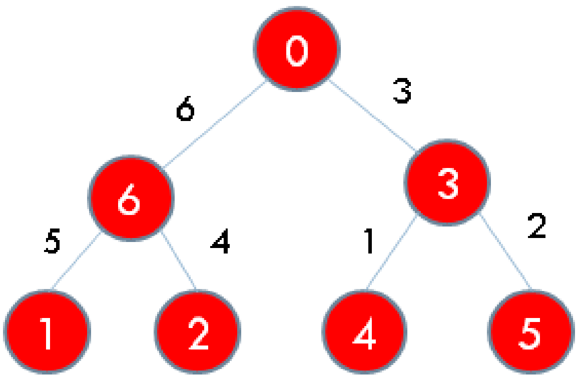


**Fig 1: An example of graceful labeling of a tree**

The Ringel-Kotzig conjecture that all trees are graceful has been the focus of many papers [2]-[8], [11], [13]-[16], [18], [20], [22]. Many classes of trees have been proven graceful in the attempt to prove that all trees are graceful. Many classes of trees have been shown to be graceful [18]. Initially, the gracefulness of several classes of trees was established by Rosa in [19]. Since then, other classes have been shown to admit graceful labeling. This referred paper [11] is contain a great source for finding a list of graceful classes of trees. Among them some of the trees which are known to be graceful are: paths, caterpillars, symmetrical trees, spider trees, lobsters, star, firecrackers, banana trees etc. Trees of diameter at most five and trees with up to 35 vertices have also been shown graceful. Yet, knowing all of them is still not enough to conclude that all trees are graceful.

Graceful labeling of graphs, where the vertices are assigned values subject to certain conditions, have often been motivated by practical problems. Moreover, Gracefully

Labeled graphs/trees are finding applications in Coding theory, Radar, Astronomy, Circuit design, Database management, Communication network addressing, X-ray crystallography etc. Therefore, Most of the researcher finds interest on proving Graceful Tree Conjecture (GTC). Our aim is to provide them a better, organized and visual survey so that they can do their research in faster and efficient way.

## 2. BACKGROUND STUDY

In the attempt to know list of all classes of graceful tree we find [11] is a great source. Yet knowing all of them is still not enough to say that all trees are graceful. In this section, we are trying to exhibit a lot of those classes in a brief manner. A **path** is a a tree in which all vertices have degree *0* or *1*. A **caterpillar** is a tree such that if one removes all of its leaves, the remaining graph is a **path** [19]. Let $T_0$ be any arbitrary caterpillar and $T_i$, $i = 1, ... , k$ be caterpillars with $|T_i| = m$ number of vertices and sum total of vertices's is the same in odd levels of all pairs $T_{2i+1}$ and $T_{2i+2}$. In case $k$ being an odd number, one caterpillar will be without a pair. Let one end of each backbone be joined to the vertex $v$ by an edge. Then the resulting tree is called a **super-caterpillar**. Let there be an even number $kp$ caterpillars, each having $m$ vertices's and sum total number of vertices's in odd (or even) levels of those caterpillars are the same. These caterpillars are grouped in $k$ groups each having $p$ caterpillars. Let backbones of the group $i$ of caterpillars be connected to a vertex $v_i$ that is connected to vertex $v$. Then the resulting tree is called a extended super-caterpillar [13]. A **symmetrical tree** is a rooted tree in which every level contains vertices of the same degree. A **spider tree** is a tree with at most one vertex of degree greater than *2* [18]. If such a vertex exists, it is called the branch point of the tree. A leg of a spider tree is anyone of the paths from the branch points to a leaf of the tree. A **lobster** is a tree such that if you remove all of its leaves, it becomes a caterpillar. A **firecracker** *F* is a tree consisting of a path *P(F)* and a collection of stars, where each vertex on *P(F)* is joined to the central vertex of exactly one star [9], [10]. A **regular bamboo tree** is a rooted tree consisting of one cetral vertex, and several legs of equal length attached to it, the leaves of which are identified with leaves of stars of equal size [21]. A **banana tree** consists of a vertex $v$ joined to one leaf of any number of stars [6]. A tree with one internal node and $k$ leaves is said to be a **star** $S_{1,k}$ that happen to be a complete bipartite graph $K_{1,k}$. A **Superstar** is a tree that consists of several stars all connected to a single star by sharing their leaves [16]. A *m*-**Star** has a single root node with any number of paths of length $m$ attached to it. An **olive tree** $T_k$ is a spider tree with $k$ legs with lengths *1, 2, ..., k* respectively [1] . A **spraying pipe tree** is a path $v_1, v_2, ... , v_n$ such that each vertex $v_i$ is joined to $m_i$ paths at a leaf of each path, and all paths have the same length [9]. A **coconut Tree** *CT(m, n)* is the graph obtained from the path $P_n$ by appending $m$ new pendent edges at an end vertex of $P_n$ [17]. A class of tree called $T_P$-trees (transformed trees) are created by taking a gracefully labeled chain and shifting some of the edges [12]. Among all of them example of some classes of gracefully labeled trees are illustrated in (see Figure 2, Figure 3, Figure 4, Figure 5 and Figure 6). We have studied all the algorithm needed to label above mentioned classes of graceful tree so that we can visualize all classes of trees by implementing all the algorithm in the back end.

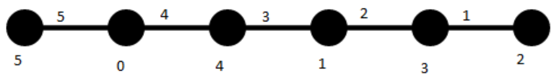


**Fig 2: An example of gracefully labeled path**

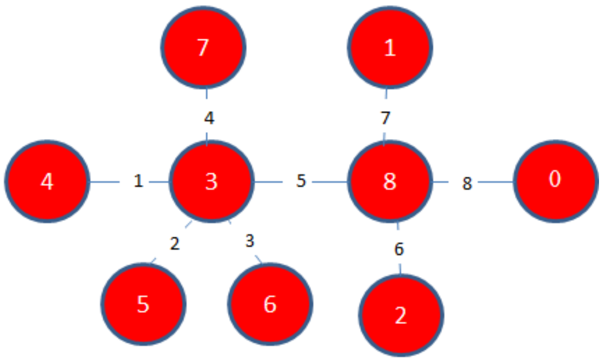


**Fig 3: An example of gracefully labeled caterpillar**

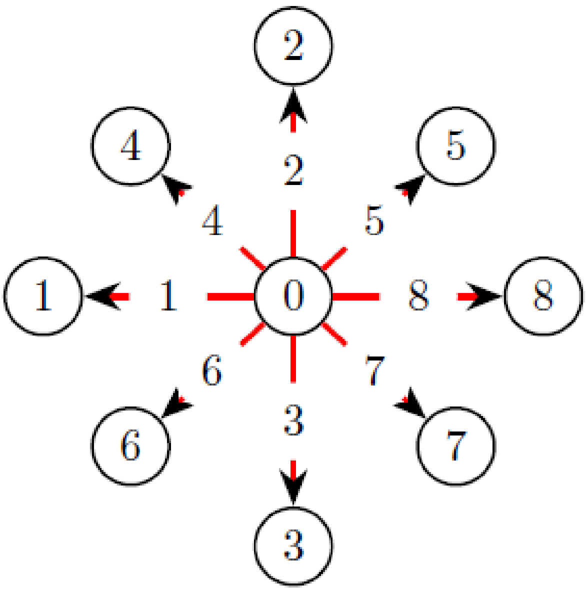


**Fig 4: An example of gracefully labeled star**

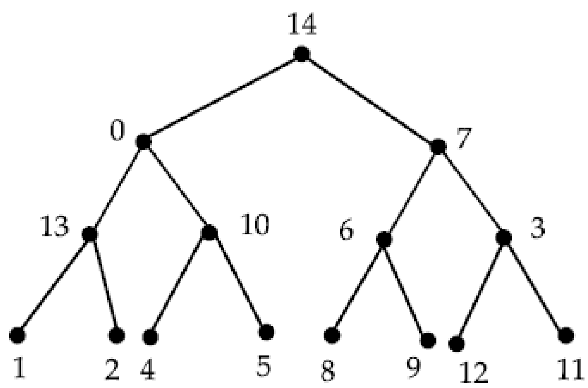


**Fig 5: An example of gracefully labeled symmetrical tree**

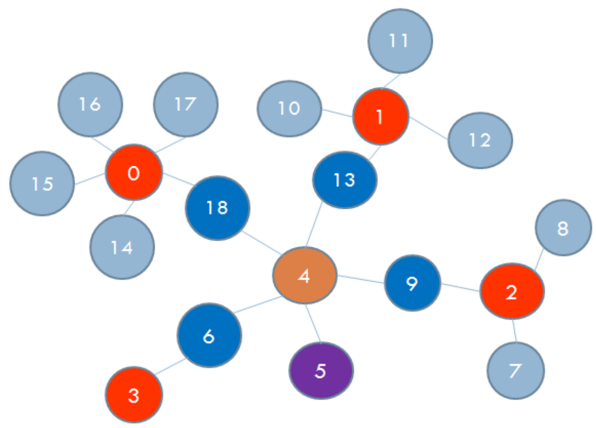


**Fig 6: An example of gracefully labeled superstar**

## 3. FEATURES OF VGTOOL

We are focusing on Graceful labeling of trees. The idea behind this web tools comes up in our mind during the research work on graceful labeling of trees where we were trying to find out new classes of graceful tree. We were thinking if there is a simplest visualize tool where we can find out whether this class of tree have already been proved to be graceful or not then it will be easy for us. After that we have decided let's make this idea works in reality.

Our idea are like there will be a on-line portal named VGTool where there will be sign in option for users by providing some basic information like name, email address, occupation, cell no, user type (General/Researchers) etc. In the portal there also be a options like FAQs and contacts for quick assistance of all users. In the home page of the portal there will be brief description about graceful labeling so that the user who do not have any idea regarding graceful labeling can shed some light on. A sample screen-shots of the portals home page is shown in (see Figure 7).

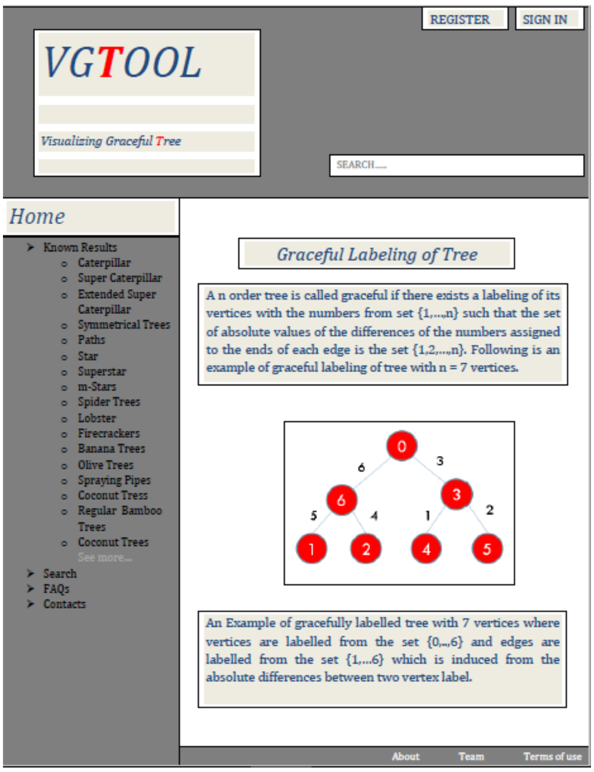


**Fig 7: Screenshots of portals home page**

In the portal there will be a list of known results on gracefully labeled tree and there will be a search option to identify inputted tree and its class or whether this tree has been proved to be gracefully labeled or not.

In the list of known results on gracefully labeled tree, for each class of tree there will have a brief description of that particular tree and there will be a option like enter sample tree by providing vertex and edge information. In this option user need to give proper tree information and then need to press simulate button. After pressing simulate button the web tool will visualize the tree with graceful labeling and this visualization will works in two phase. In the portal we will only see the visualization but in the back end system will implement appropriate algorithm to label the inputted tree. A sample screen-shots of the portals known results page and search under known results page are shown in (see Figure 8 and Figure 9).

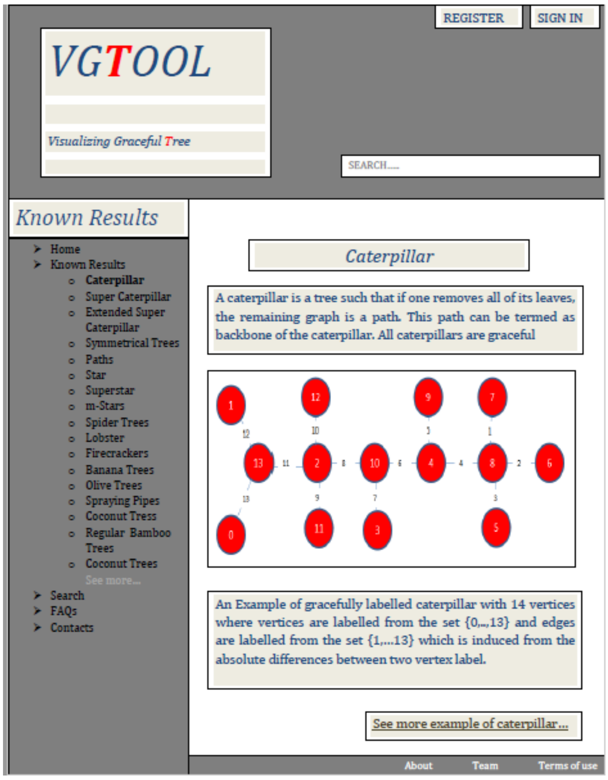


**Fig 8: One of the screenshots of portals known results page**

Search option will be used by the researcher who don't know the actual class of tree they are thinking. In the search option there will be another option to enter users/researchers random tree. After giving input there will be two option one is to check class of the inputted tree and another one is to check gracefulness of the inputted tree. In each option web tool will show the message regarding status of inputted tree like graceful or a class name of gracefully labeled tree from known results like caterpillar or not yet discovered to be gracefully labeled. This part is the most crucial part of our work. As because its not easy to say for any random tree its not gracefully labeled tree. Our idea is to match inputted tree pattern with known results pattern if it matches with any known results pattern then we can easily say the class name of the inputted tree but if not we will say not yet discovered the inputted tree is graceful. Therefore researcher will get a clear conception whether they should work on this tree or not. A sample screen-shots of the portals search page is shown in (see Figure 10).

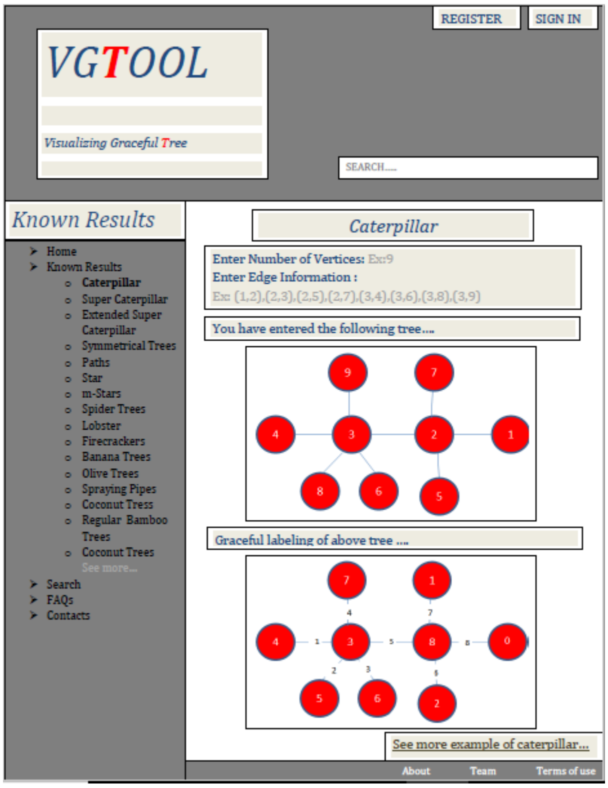


**Fig 9: Another screenshots of portals known results page**

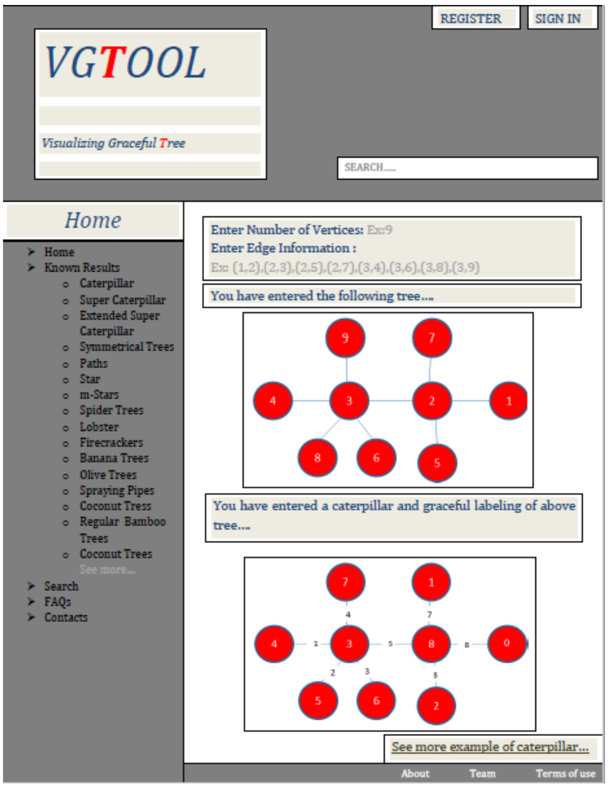


**Fig 10: Screenshots of portals search page**

# 4. EXPERIMENTAL RESULTS

For known results there are different algorithm for different classes of graceful tree. In the attempt to know all classes of gracefully labeled tree we have studied almost all the algorithm which is used to label this tree. Therefore we have decided to implement all those algorithm for each category of trees in the background of the portal. The details of the experiment we have done for implementing algorithm for superstar are explained in the following sections.

## 4.1 Device

The codes implemented for our research were developed in a device running of Window OS. The minimum requirement for the example demonstration is the installation of Flash Player (minimum version 11). An update web browser will help in that case. The codes can be executed smoothly in Windows or Linux based devices.

## 4.2 Language

The language used for the development of the samples provided with this paper is ActionScript 3.0. It is an object-oriented programming language which was originally developed by Macromedia Inc., later dissolved by Adobe Systems. It is a superset of the syntax and semantics of the language JavaScript. The language is primarily used for the development of websites and software targeting the Adobe Flash Player platform.

## 4.3 IDE

The IDE used for the development of the algorithms and the example codes provided with this paper was Adobe Flash Professional CS6. It is a part of the Adobe CS6 Master Collection. Adobe Flash Professional is a multimedia authoring program used to create content for Adobe Engagement Platform. It is used to develop web applications, games, movies, content for mobile phones and other embedded devices. The platform supports the scripting language ActionScript 3.0 in case of user interaction and graphical manipulation. The version, Adobe Flash Professional CS6 as released in 2012. It was upgraded from the previous versions by integrating the support of HTML5 and the ability to generate spread sheets.

## 4.4 Code Specification for Superstar

In order to execute the algorithm, two integer variables has to be provided to the function. The number of arms for each star will be provided as armsNumber1 and armsNumber2 respectively, where armsNumber1 is a positive integer and armsNumber2 is a non-negative integer. After the simulation, the vertexNumber array will contain the labels of nodes in graceful manner. The sequence will be as follows:

- Initial index (0) will contain label of the center node of first star
- Following armsNumber1 indices will contain the labels of the arms of the first star
- The next index will contain the label of the center node of the second star
- Following armsNumber2 indices will contain the labels of the arms of the second star
- The last index, i.e. the index indicating the number of vertices will contain the label of the joining node ($n^{th}$ vertex).

Apart from the array, two variables join1 and join2 will contain the edge numbers of the edges joining the $n$th vertex with the observed arms of the first star and second star respectively.

### 4.5 Input Format for Superstar

The input for the simulation of the Superstar are integer number. The first input textbox is for the number of arms of the first star, which must be a positive integer. The second textbox is for the number of arms of the second star. Pressing simulate will provide the required graphical illustration of the gracefully labeled Superstar. In the provided test case, the inputs were 5 and 7 in (see Figure 11).

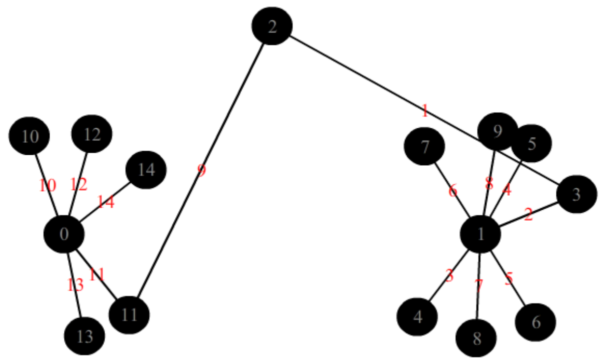


**Fig 11: An example of gracefully labeled Superstar**

## 5. CONCLUSION AND RECOMMENDATION

The system have designed and proposed a framework specially for a researcher who want to know little bit more about graceful labeling. It will help user to discover new classes of graceful trees or verify existing classes of graceful trees with new classes of graceful trees. It also helps user to understand how graceful labeling actually works. Though the idea is in very early state but it could be possible to make it more interactive and fruitful in the coming age by enhancing the feature and properties of the portals and adding more results of graceful tree.

## 6. REFERENCES

[1] Abhyankar, V. J., and Bhat-Nayak, V. N. 2000. Easiest graceful labeling of olive trees. Bull. Bombay Math. Coll., 14:16–25.

[2] Abhyanker, V. J. 2002. Direct methods of gracefully labelling graphs. Technical report, University of Mumbai.

[3] Alfred, R. and McKay, B. 1998. Graceful and harmonious labeling of trees. Bull. Inst.Appl., 23:69–72.

[4] Alfred, R., Siran, J. and Siran, M. 2003. A note on the number of graceful labelings of paths. Discrete Math. 261:27–30.

[5] Bermond, J. C. 1979. Graceful graphs, radio antennae and french windmills. Graph Theory and Combinatorics, Pitman, London, 18–37.

[6] Bhat-Nayak, V. and Deshmukh, D. 1996. New families of graceful banana trees. Proc. Indian Acad. Sci. Math. Sci. 106:201–216.

[7] Bloom, G. S. and Golomb, S. W. 1977. Applications of numbered undirected graphs. Proceedings of the IEEE. 165(4):562–570.

[8] M. Burzio and G. Ferrarese. 1998. The subdivision graph of a graceful tree is a graceful tree. Discrete Math. (181):275–281.

[9] Chen, W. C., Lu, H. I. and Yeh, Y. N. 1997. Operations of interlaced trees and graceful trees. Internat. J. Algor. Comput. Math. 21:337–348.

[10] Edwards, M. and Howard, L. A. 2006. survey of graceful trees. Atlantic Electronic Journal of Mathematics. 1(1).

[11] Gallian, J. 2014. A dynamic survey of graph labeling. the electronic journal of combinatorics. 17.

[12] Hegde, S. M. and Shetty, S. 2002. On graceful trees. Applied Mathematics E-Note. (2):192–197.

[13] Hossain, M. F., Aziz, A., Momin, M. and M. Kaykobad. 2014. New classes of graceful trees. Utilitas Mathematica. 2014:6.

[14] Huang, C. K. and Rosa, A. 1982. A further results on tree labellings. Utilitas Mathematica, 21:31–48.

[15] Jin, D. J. and Wang, J. 1993. The gracefulness of the tree with a diametertour. Technical report, Acta Scientiarum Naturalium Universitaties Jilinensis,1:17-22.

[16] Munia, A. A., Maowa, J., Tania, S. and Kaykobad, M. 2014. A new class of graceful tree. International Journal of Scientific and Engineering Research (IJSER), 5(11):1112–1115.

[17] Ramachandran, V. and Sekar, C. 2014. One modulo n gracefulness of regular bamboo tree and coconut tree. International Journal on Applications of Graph Theory in Wireless Ad hoc Networks & Sensor Networks, 6(2).

[18] Robeva, E. 2011. An extensive survey of graceful trees. Technical report. Stanford University.

[19] Rosa, A. 1967. On certain valuations of the vertices of a graph. Inter-nat. Symposium, Rome, July 1966, 349–355.

[20] Roy, T. G., Aryabhatta, S., Uddin, M. M. and Rahman, M. S. 2011. Ongraceful labelings of trees. In Proceedings of WALCOM, Lecture Notesin Computer Science, 6552, 17:214–220.

[21] Sekar, C. 2002. Studies in graph theory. Technical report, Ph.D. Thesis, Madurai Kamaraj University.

[22] Watson, R. L. 1972. A survey on the graceful labeling of graphs. Technical report, B.S., Roanoke College.