**BANSILAL RAMNATH AGARWA CHARITABLE TRUST’S**

**VISHWAKARMA INSTITUTE OF INFORMATION TECHNOLOGY,PUNE**

**DEPARTMENT OF COMPUTER ENGINEERING**

**WORK BREAKDOWN STRUCTURE(WBS)**

GROUP ID:- S.Y.B.TECH

STUDENTS:-

MEGHA CHANDWANI -221015

SAKSHI CHORDIA -221020

PARJANYA DESHMUKH -221025

HRISHIKESH DATE -221021

TOPIC:- STUDENT DATABASE

In our project we have made a binary tree which helps to insert,search and edit the details of the student based on his/her division, department and roll no. and gives fastest and the shortest way with sorted and well balanced Binary Tree. . Thus inorder traversal is time- efficient and also easy to code.

We are able to access the data of any student by mentioning his/her Roll-No Department And Division.

We have done this with the use of classes thus keeping the OOP principle of Data Hiding intact

We are able to edit the data of any student by entering the student details.

Every Key of the Node in the binary tree is the student Registration Number and hence that cannot be edited. Sorting of elements is based on the Registration Number.

Each node contains-

1-Student Reg Number

2-Department

3-Divison

4-Telephone Number

5- Date of Birth

6-Email-ID

The data structure is we used is a BINARY TREE.

The advantage of a Binary Search is the growth factor of the search as you add more items to the container (Binary Tree)

Searching a sorted, well-balanced Binary-Tree has a run-time of O(log(n)),

that log would have a base of 2.

In comparison, simply searching through the entire collection of items would have a run time of O(n)

It means that the well-balanced (and sorted) Binary-Tree can keep growing and growing and growing, and as the number of items (‘n’) doubles, the number of comparisons (a.k.a. work) goes up by one unit (at worst). While going through a list, checking each item you come across will increase the amount of work by one unit per item

It is one of the fastest way to search an item, the performance is Log N, so it finds in an array of 1 mln elements with only about 20 comparison. Also only 64 comparison on 2^64 elements, which is really big number

Our Basic Structure is shown below -

Reg Number

Department , Division

Telephone-Number, Date Of birth ,Email-ID

Reg Number

Department Division

Telephone-Number, Date Of birth ,Email-ID

Reg Number

Department , Division

Telephone-Number, Date Of birth ,Email-ID

