**Distributed Hash Tables**

**Project Report**

**“Team”**

**Hamza Khalid - 19I-2011**

**Hamza Iftikhar - 19I-2003**

**Muhammad Zahid - 19I-0496**

**Project Source Files:**

Main.cpp

SHA1.cpp

DHT.cpp

AVL.cpp

FTP.h

SHA1.h

DHT.h

AVL.h

FTP.h

**Implementation:**

User is asked to enter the number for bits for the DHT. Then the user specifies the number of machines for the DHT.

The DHT is a circular linked list that has nodes equal to 2bits-1. The Machines in the DHT is a singly linked list.

**Machine:**

Each Machine in the DHT stores data of it’s successors in, For Example if a Machine is at ID 4 and another Machine is at ID 7, then Machine 7 will store the data for the IDs 5 6 7.

**Data Storing:**

To Store the Data, the User either enters just the value or Key value pair to store in the machine. The Data entered by the user is hashed by the SHA1 algorithm and then sent over to the DHT to store in the Machine. The DHT function first figures out which machine to store the data in. If the machine is not specified by the user then the Hashed Key is used to Generate an ID. The Data is then store at that location.

**File:**

The Data in Each file is handled by an AVL tree. Each AVL node holds the key and the line number at which the data is stored.

**Data Retrieval:**

To Get the Data, The Routing Table is used to arrive at the Machine. Then the File’s AVL tree is traversed to look for the key if it exists. If the key is found in that AVL. The file is loaded onto a Linked-List. The line number from the AVL node is read and the data is return by using the get(line number) function of the linked list.

**Machine Addition:**

If the user does not Specify the Machine ID, An automatic Machine ID is generated. Now For example the ID generated is 5. Then the Data from Machine 7(Which holds the Data for ID 5) is used to get the Data for Machine 5. Each Key in the AVL of Machine 7 is sent to the same function that generates ID to store the data. If that ID matches Machine ID 5. then it is loaded onto a Linked List. The data of that key is loaded onto another Linked List. After the files have been searched for the data. The newly made Linked Lists are now used to insert that data in the Machine ID 5. After this, the machine Addition is complete. The Keys loaded onto the Linked List are then used to delete the data and the nodes from machine ID 7. since we no longer need them.

**Machine Deletion:**

Now Suppose we want to delete the Machine we just added into our DHT. We want to delete Machine ID 5. To do that, we grab all the data from that machine and insert it to the machine that comes after it, In this case Machine ID 7. then we delete that Machine. After that Machine Deletion is complete.

**Finger-Tables:**

Finger tables are implemented as doubly linked list using the

formula FTp[i]=succ(p+2i-1) given in the project manual. To create finger tables we have made a function which has to be called after the ids are assigned to each machine in the DHT. After a machine is inserted or deleted finger tables are updated accordingly.

