**Project-4**

**Perfect hash table based Telephone Directory**

**Abstract:** This project demonstrates prefect hash table and its applications. Student has to implement the following tasks - A hash table for integers and integers with a universal hash function, A perfect hash table for integers, sort parallel arrays, count the number of times each word appears, Merging Ordered Lists, Threaded Binary Tree ADT

**Module: 1**

**A hash table for integers with a universal hash function.**

1. Function: get\_node()

2. Function: return\_node

3. Function: create\_hashtable

4. Function: universalhashfunction

5. Function: find

6. Function: insert

7. Function: delete

8. Function: find\_mtf

9. Function: list\_table

10. Function: main()

**Module: 2**

**A hash table for strings with a universal hash function.**

1. Function: get\_node()

2. Function: return\_node

3. Function: create\_hashtable

4. Function: universalhashfunction

5. Function: find

6. Function: insert

7. Function: delete

8. Function: find\_mtf

9. Function: list\_table

10. Function: main()

**Module: 3**

**A perfect hash table for integers.**

1. Function: create\_perf\_hash

2. Function: find

3. Function: main()

4. Function: compute secondary table sizes and their offset

5. Function: mark bucket as defect

6. Function: secondary hash tables

**Module: 4**

1. Rewrite insertionSort3 to illustrate how to sort parallel arrays: Each time a name is moved

during the sorting process, the corresponding ID number must also be moved. Since the name and ID number must be moved “in parallel,” we say we are doing a *parallel sort* or we are sorting *parallel arrays*.

2. Write a program to read an English passage and count the number of times each word appears.

The output consists of an alphabetical listing of the words and their frequencies. This is a typical

“search and insert” situation.

3. Merging Ordered Lists: Merging is the process by which two or more ordered lists are combined into one ordered list.

4. Threaded Binary Tree ADT

5. Function: Insert

6. Function: Delete

7. Function: Search

8. Function: In order traversal