**DOKUZ EYLUL UNIVERSITY**

**ENGINEERING FACULTY**

**DEPARTMENT OF COMPUTER ENGINEERING**

**CME2002 – Data Organisation and Management**

**ADDRESS BOOK APPLICATION WITH SIMPLE INDEXING**

**by**

**Özgür Hepsağ**

**May, 2017**

**İZMİ****R**

INTRODUCTION

Aim of this assignment is to learn binary search and indexing. In this assignment, we write our records into data file and keep our primary keys and indexes into index file. These files must be binary files to read and write properly. There are some rules to synchronise these two files. First rule is that records are always added to end of the data file and correct index is given. Then these recors are kept sorted in RAM with primary(name and surname) key and index. When program was closed succesfully, index file is created. If program was closed in an unexpected way, program will recreate the primary keys and index from data file again.

## ASSIGNMENT DESCRIPTION

Our address book application has menu and you can choose some operations. These operations are “Add New Record, Update A Record, Delete A Record, Find A Record, List Records, Exit”. I will explain all this operations and how working this program. When program is open, check firstly dirty bit from index file (if index file exists). If dirty bit is active, program read all records from data file and re-index all of them and keep them RAM. If dirty bit is not active, program read index file and keep them in RAM. Then, our menu will appear. You can select whatever you want.

**Add New Record:** Program wants to be entered inputs (name, surname, address, addition note) one by one. But, the user cannot enter multiple words with spaces. This is the disadvantage of this my program. After that, program checks previous primary keys and control if new record is same with old records. If not same with old records, new data are added end of the data file.

**Update A Record:** Program wants to be entered only single character firstly. Following, records are listed starting with this character. Then, the user should select record that which one he want to update and enter new values. Finally, new record is created and overwrites into the data file on old data. Program can find easily the location of the file with index of that.

**Delete A Record:** Program wants to be entered name and surname. Then, index of that record are found from RAM for location of record in data file and “\*” are written 30 bits (equals name of the record) on data file.

**Find A Record:** Firstly, all records are listed as for that single character. This operation is easily done by indexes in RAM. Then, Program wants to be entered name and surname and find index of that from RAM and locate in data file and finally read the all of the record.

**List Records:** Program wants to be entered only single character firstly. After that, records are listed starting with this character. Records are easily read from data file with indexes.

**Exit:** When user select exit, dirty bit is changed and index file are written

## PROBLEMS ENCOUNTERED

First problem that I have encountered is, if there is not data file, my code does not create data file. I can fix that problem with if there is not data file program, open the data file “wb” mode or if there is data file, program open data file wit “rb+” mode.

Second problem is that scanf() function get only 1 word. But, there can be multiple word on address or addition note field. I could not handle this problem. I asked from user enter “-” between words instead of space.

## CONCULUSION

The program works correctly and all operations are done successfully. Consequently, assignment is fully done and aim of this assignment is accomplished.