International Final University

Computer/Software Engineering Department
Fall 2022
COMP215/SOFT215 Algorithms and Data Structures

ASSIGNMENT 2

Purpose : To perform basic operations on linked list structures using C Language

Due date :
Submission of the report on word on LMS (23/1/23 up to 9:00 PM),
Program Grading During Lab Hour 24/1/2023

Definition of the Problem :

Basic operations are **Insertion, Deletion , Searching and Sorting** using linked list data structures.

Telephone directory system of the subscribers have the following information fields.

Telephone number, Name , Last name.

Structure of the "telephone.txt" file has the following fields

Telephone Number(7)
Name(12)
Last Name(12)

You may define your node structure as follows

```
struct  tlpInfo{
        long telNumber;
        char  firstName[12];
        char  lastName[12];
        };
struct node {
        struct tlpInfo  info;
        struct  node *link;
        };
typedef  node *NODEPTR;
```

Write a menu driven C program and create a simple sorted linked list data structures which will have the node structures defined as above and do the following operations:

1. (**10 Points**)Read from the file "**telephone.txt"** and **create** a linked list
    - Display Linked list structure. 30 Person on each page.
2. **(15 points)**Read from the file "**telephone.txt"** and **create** a **Sorted** linked list structure (using **telephone** Number in **ascending order**). Use insertion sort method
    - **Display sorted list structure from the monitor**(Telephone number, name, last name of the subscriber), 30 Person on each page.

3. **(15 Points)Sort by name in ascending order**(Using same insertion sort method again)) .
    - **Display**(sorted base on name) subscriber information(Telephone number, name, last name of the subscriber), 30 Person in each page. Use string function

4. **(15 points)**.**New Subscriber Insertion**

   Read a new Subscriber information (Telephone number, name, last name) as a user's keyboard input, and **Insert** his/her information into the proper location in the sorted linked list structure. Afterwards,
    - Display all subscriber information based on telephone numbers). Before insertion, linked list structure must be sorted based on telephone numbers.

5. **(15 points) Search a person using a given name and last name and list subscriber information**

   Read a name , last name from the keyboard and using sorted linked list information based on name.
    - Search the person from the sorted linked list and find and display subscriber information. Give error message if person is not in the linked list structure. Duplicated names will be checked using last names.
6. **(15 points)Search and Delete**

   Read a subscriber information(Telephone number) from keyboard , find and **Delete** corresponding subscriber from the sorted linked list structure . After confirmation (yes/no). Check all necessary conditions. If given subscriber information is not in the linked list, your program must give an error messages and repeats the input .

   After deletion,
    - **Display subscriber** information 30 Person on each page.

USER CATALOGUE WILL BE PREPARED WITH THE FOLLOWING HEADLINES:

- **Definition of the problem:** Given problem will be explained.

- **User documentation**: How can a single user run your program? Describe in details.

- **Input environment:** What does your program get as an input of information? Explain the content of your input files.

- **Output environment**: What does your program produce? Production of your program will be explained.

- **Software design issues:** Describe the main program and functions (if any) your program.
- **Program list:** List of the program and Input/Output files.

- **Programming time**: How long did you spend to prepare this program for the following steps.

Analyze and design :
Coding :
Debugging :
Documentation :
Total :

IMPORTANT NOTE:

- SUBMISSION OF THE REPORTS WILL BE ON **MS WORD DOCUMENT FORMAT** AND WILL BE LOADED ON LMS(23 January 21:00) AND YOUR PROGRAM CHECKING WILL BE DONE BY THE INSTRUCTOR DURING LABARATORY HOURS.

Example Program:

Following program creates a linked list structure using given file “telephone.txt” and list linked list structure at the same time.

```
#include<stdio.h>
#include<string.h>
#include<conio.h>
#include<stdlib.h>

struct  tlpInfo{
          long    telnumber;
          char  firstName[12];
          char  lastName[12];
          };

struct node {
          struct tlpInfo  info;
          struct  node *link;
          };

typedef  node *NODEPTR;

NODEPTR getnode();
void fileIntoLinkedList(NODEPTR);
void display(NODEPTR);

int main()
{
NODEPTR head,p,save;
 head=NULL;
 fileIntoLinkedList(head);
 display(head);
 return 0;
}
```

```
void display(NODEPTR head)
{
NODEPTR save;

save=head;
while(save!=NULL)
{
   printf("%ld %s %s\n",save->info.telnumber,save->info.firstName,save-
>info.lastName);
   save=save->link;
}
}
void fileIntoLinkedList(NODEPTR head)
{
        struct tlpInfo temp;
        NODEPTR save,p;
        FILE *a=fopen("telephone.txt","r");

while(fscanf(a,"%ld %s %s\n",&temp.telnumber, temp.firstName, temp.lastName)!=EOF)
{ p=getnode();
  p->info=temp;
  printf("%ld %s %s\n",temp.telnumber,temp.firstName,temp.lastName);
  if (head==NULL){head=p;
              save=p;
                                }
                 else{save->link=p;
                         save=p;
                             }
}
  p->link=NULL;
}
NODEPTR getnode()
{
        NODEPTR q;
        q=(NODEPTR)malloc(sizeof(struct node));
        return q;
}
```