Programme Name: BCS-IT

Course Code: CSC-2516

Course Name: Data Structure and Algorithm

Submitted By:

Student Name: Manish Neupane

IUKL ID: 041902900037

Semester: 4th semester

Intake:2019

Submitted TO:

Faculty Name: Prakash Chandra

Department: BCS-IT

1. Abstraction :

This project is an approach to create real-life student management system that contains student's details like name , email , roll number, contact and their courses id implemented using the basic knowledge of operations on linked list like insertion , deletion , searching and showing as well as basic file I/O handling has been used to insert new data in file , update existing data of file , delete data from and search the data inside the file.

2. Introduction :

To understand this project in detail first we should know about what is student management system , what we do in this systems as well as the most important part is about linked list and file handling that has been used to make this system come in to real use .

Student management system is a software that is designed to track , maintain and manage all the data generated by educational institutes including grades of students and all their activities records. This system is used to store all the day-to-day school operations, manage and handle administrative process such as admission , billing payments , record of student information , reporting , and tracking fess ,managing administrative department of institutes etc.

But this project is capable of recording , searching, deleting , updating the general student information with the help of linked list algorithm and file handling process.

Linked is a linear data structure which consists of nodes where each node contains data field and reference to next node in the list.

file handling is the storing of data a in a file using a program .

So all the working mechanism of linked and file handling process has been explained in the feature section.

3. Feature :

The main backend working principal of this system is linked . I have created structure named as StudentInfo which act as linked list node that is able to store student details like name, address, email , roll number , contact number and courses id along with reference of another linked list . Similarly I have used same file handling concept to store data in file using the linked list.

Below are the list of operation we can perform in this system:

- Inserting the details :
  The system is able take student details input from the terminal and store that data inside the linked list and same linked list has been used to write the details inside the file in the form of stack i.e., like pile , the new data will appear in the top of list.
  The time complexity to insert element inside the linked list is O(1).
- Searching :
  To search the data program fetch the data from the file .If the file is empty, it shows empty message otherwise it stores data inside the linked and show the data to the users.
  The searching is based on the roll number of student .
  The searching has been implement using binary search algorithm in linked list .
  The time complexity of searching depends on the number of element present inside the linked list i.e., O(n).
- Deletion :
  Deletion is also based on the roll number of student. To delete the data program, fetch the data from the file .If the file is empty, it shows empty message otherwise it stores data inside the linked and the program searches the given roll number is present inside the list or not . If present inside file it retrieves data inside the linked list and delete the data from the file.
  The time complexity of deleting depends on first searching the elements on the basis which is same the time complexity of searching feature and removing the element from the list is always 1.
  Updating :
  Updating is also based on the roll number of student. To update the data program, fetch the data from the file .If the file is empty, it shows empty message otherwise it stores data inside the linked and the program searches the given roll number is present inside the list or not . If present inside file it retrieves data inside the linked list allowing the user to enter new data and update the data from the file .
  The time complexity of updating depends on first searching the elements on the basis which is same the time complexity of searching feature and updating the whole element of the list is same like inserting feature .

4. Testing

inserting data inside the linked list

```
Do you want to continue(y/n)?
y

Press :
i.To Insert the data .
u. To Update the data .
d. To Delete the data.
s. To Search the data.
p. To print data on terminal
i

Enter the name :
pravesh

Enter the rollno :
43

Enter the address of student :
baneshwor

Enter the email of student :
praveshsingh13@gmail.com

Enter the contact number of student:
9847057010
Enter the Cousrse Id of 1 subject : 3201
Enter the Cousrse Id of 2 subject : 3200
Enter the Cousrse Id of 3 subject : 2516
Enter the Cousrse Id of 4 subject : 2415

Do you want to continue(y/n)?
```

```
PS C:\Users\HP\Desktop\individual project work\DAS\dsa> ./a

                        ==========STUDENT MANAGEMENT SYSTEM========


Press :
i.To Insert the data .
u. To Update the data .
d. To Delete the data.
s. To Search the data.
p. To print data on terminal
i

Enter the name :
manish

Enter the rollno :
12

Enter the address of student :
tinkune

Enter the email of student :
neupanemanes@gmail.com

Enter the contact number of student:
9810438054
Enter the Cousrse Id of 1 subject : 2624
Enter the Cousrse Id of 2 subject : 2116
Enter the Cousrse Id of 3 subject : 3200
Enter the Cousrse Id of 4 subject : 3201
```

```
Press :
i.To Insert the data .
u. To Update the data .
d. To Delete the data.
s. To Search the data.
p. To print data on terminal
i

Enter the name :
sumana

Enter the rollno :
29

Enter the address of student :
koteshwor

Enter the email of student :
sumanarai@gmail.com

Enter the contact number of student:
9880402099
Enter the Cousrse Id of 1 subject : 3201
Enter the Cousrse Id of 2 subject : 3200
Enter the Cousrse Id of 3 subject : 2516
Enter the Cousrse Id of 4 subject : 2117
```

Writing data in the file :

```
12 manish tinkune neupanemanes@gmail.com  9810438054 2624 2116 3200 3201
29 sumana koteshwor sumanarai@gmail.com  9880402099 3201 3200 2516 2117
43 pravesh baneshwor praveshsingh13@gmail.com  9847057010 3201 3200 2516 2415
```

Searching :

```
Press :
i.To Insert the data .
u. To Update the data .
d. To Delete the data.
s. To Search the data.
p. To print data on terminal
s
inside the search method
Enter the roll number of student :
43
name = pravesh
Address =baneshwor
Contact number = 9847057010
E-mail = praveshsingh13@gmail.com
cousrse id of subject 1 = 3201
cousrse id of subject 2 = 3200
cousrse id of subject 3 = 2516
cousrse id of subject 4 = 2415
==========================================
```

```
Press :
i.To Insert the data .
u. To Update the data .
d. To Delete the data.
s. To Search the data.
p. To print data on terminal
s
inside the search method
Enter the roll number of student :
12
name = manish
Address =tinkune
Contact number = 9810438054
E-mail = neupanemanes@gmail.com
cousrse id of subject 1 = 2624
cousrse id of subject 2 = 2116
cousrse id of subject 3 = 3200
cousrse id of subject 4 = 3201
==========================================
```

```
Press :
i.To Insert the data .
u. To Update the data .
d. To Delete the data.
s. To Search the data.
p. To print data on terminal
s
inside the search method
Enter the roll number of student :
29
name = sumana
Address =koteshwor
Contact number = 9880402099
E-mail = sumanarai@gmail.com
cousrse id of subject 1 = 3201
cousrse id of subject 2 = 3200
cousrse id of subject 3 = 2516
cousrse id of subject 4 = 2117
==========================================
```

Deleting :

```
Press :
i.To Insert the data .
u. To Update the data .
d. To Delete the data.
s. To Search the data.
p. To print data on terminal
d

Inside the delete method :

Enter the roll number of student :
43
Record with roll number 43 Found !!!
Record Successfully Deleted !!!

writing  data inside the file :
```

student.csv

```
29 sumana koteshwor sumanarai@gmail.com 9880402099 3201 3200 2516 2117
12 manish tinkune neupanemanes@gmail.com 9810438054 2624 2116 3200 3201
```

```
Press :
i.To Insert the data .
u. To Update the data .
d. To Delete the data.
s. To Search the data.
p. To print data on terminal
d

Inside the delete method :

Enter the roll number of student :
29
Record with roll number 29 Found !!!
Record Successfully Deleted !!!

writing  data inside the file :
```

```
12 manish tinkune neupanemanes@gmail.com 9810438054 2624 2116 3200 3201
```

Updating :

```
                    ==========STUDENT MANAGEMENT SYSTEM===========


Press :
i.To Insert the data .
u. To Update the data .
d. To Delete the data.
s. To Search the data.
p. To print data on terminal
u

inside the update method

Enter the roll number of student :

12
Record with roll number 12 Found !!!

 Enter new name :anisha

Enter the address of student :
balkumari

Enter the email of student :
anish12@gmail.com

Enter the contact number of student:
9810756912

updation sucessfull

writing  data inside the file :
```

sa > student.csv

```
1   43 pravesh baneshwor praveshsingh13@gmail.com 9847057010 3201 3200 2516 2415
2   29 sumana koteshwor sumanarai@gmail.com 9880402099 3201 3200 2516 2117
3   12 anisha balkumari anish12@gmail.com 9810756912 2624 2116 3200 3201
```

```
Do you want to continue(y/n)?
y

Press :
i.To Insert the data .
u. To Update the data .
d. To Delete the data.
s. To Search the data.
p. To print data on terminal
u

inside the update method

Enter the roll number of student :
43
Record with roll number 43 Found !!!

 Enter new name :parvin

Enter the address of student :
newbaneshowr

Enter the email of student :
pravinkumar@14gmail.com

Enter the contact number of student:
9810438054

updation sucessfull

writing  data inside the file :
```

dsa > student.csv

```
1   12 anisha balkumari anish12@gmail.com 9810756912 2624 2116 3200 3201
2   29 sumana koteshwor sumanarai@gmail.com 9880402099 3201 3200 2516 2117
3   43 parvin newbaneshowr pravinkumar@14gmail.com 9810438054 3201 3200 2516 2415
```

5. Conclusion :

The main aim of this project to implement linked list using real- life system . We have performed different operation linked list that has made possible to make student management system.
We even can enhance this system by using other implementations of linked list like queue and stack to store and retrieve data . Similarly, we can use other key value for searching ,deleting and updating data of the student details of particular data .
Overall this system is the demo on what is student management system and how it will work if it is implemented using linked list data structure . Here linked list has been used to store data to write inside data and fetching data from file to search ,delete etc. Linked list is dynamically declared so we do not have to allocate memory in advance as well as linked list increases or decrease at run-time . By the use of linked binary search has been easier as it has very fast excess time. We have implemented stack data structures to insert data inside the file .

6.Appendix :

```c
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
struct info
{
    char name[50];
    int data;
    char address[35];
    char email[50];
    char contact_num[11];
    int courseID[4];
    struct info *next;
};
struct info *datanode = NULL;
```

Inserting into node after fetching data from file:

```c
void insertintonode(char name[30], int number,char address[50],char email[50],
char contact_num[11], int 12oursed[])
{
    struct info *insertnode = malloc(sizeof(struct info));
    strcpy(insertnode->name, name);
    insertnode->data = number;
    strcpy(insertnode->address, address);
    strcpy(insertnode->email, email);
    strcpy(insertnode->contact_num, contact_num);
    for (int I = 0; I < 4; i++)
    {
        insertnode->12oursed[i] = 12oursed[i];
    }

    insertnode->next = datanode;
    datanode = insertnode;
}
```

This method help to fetch the data from the file

```
void fetech()
{
    FILE *fileptr;
    fileptr = fopen("student.csv", "r");
    if (fileptr == NULL)
    {
        printf("\n no element inside the file");
    }
    else
    {

        struct info *tempnode;
        tempnode = (struct info *)malloc(sizeof(struct info));

        char name1[30];
        char address[35];
        char email[50];
        char contact_num[11];
        int item;

        char endofflie;
        while (endofflie != EOF)
        {
            fscanf(fileptr, "%i ", &tempnode->data);
            fscanf(fileptr, "%s ", name1);
            fscanf(fileptr, "%s ", address);
            fscanf(fileptr, "%s ", email);
            fscanf(fileptr, "%s ", contact_num);
            for (int i = 0; i < 4; i++)
            {
                fscanf(fileptr, "%i", &tempnode->courseID[i]);
            }
            strcpy(tempnode->name, name1);
            strcpy(tempnode->address, address);
            strcpy(tempnode->email, email);
            strcpy(tempnode->contact_num, contact_num);
            insertintonode(tempnode->name, tempnode->data,tempnode-
>address,tempnode->email,tempnode->contact_num,tempnode->courseID);
            endofflie = fgetc(fileptr);
        }
    }
    fclose(fileptr);
```

```
}
```

This code is to rewrite the data inside the after deletion and update

```
void writeinfile(char *ch)
{

    printf("\nwriting  data inside the file :\n");

    FILE *filewrite;
    filewrite = fopen("student.csv", ch);
    struct info *writefilenode = datanode;

    while (writefilenode != NULL)
    {
        fprintf(filewrite, "\n%i ", writefilenode->data);
        fprintf(filewrite, "%s ", writefilenode->name);
        fprintf(filewrite, "%s ", writefilenode->address);
        fprintf(filewrite, "%s ", writefilenode->email);
        fprintf(filewrite, "%s ", writefilenode->contact_num);
        for (int i = 0; i < 4; i++)
        {
            fprintf(filewrite, "%i ", writefilenode->courseID[i]);
        }

        writefilenode = writefilenode->next;
    }
    fclose(filewrite);
}
```

Inserting Code :

```
void insert()
{
    struct info *insertnode = malloc(sizeof(struct info));
    char name[30], address1[50], email[50];
    int i = 0, rollno, courseId[4];
    char contact_num[11];
    int items;
    FILE *fileptr;
    fileptr = fopen("student.csv", "a");

    printf("\nEnter the name :\n");
    scanf("%s", name);
    printf("\nEnter the rollno :\n");
    scanf("%i", &items);
    printf("\nEnter the address of student :\n");
    scanf("%s", address1);
    printf("\nEnter the email of student :\n");
    scanf("%s", email);
    printf("\nEnter the contact number of student:\n");
    scanf("%s", &contact_num);
    for (i = 0; i < 4; i++)
    {
        printf("Enter the Cousrse Id of %i subject : ", i + 1);
        scanf("%i", &courseId[i]);
    }
    strcpy(insertnode->name, name);
    strcpy(insertnode->address, address1);
    strcpy(insertnode->email, email);
    strcpy(insertnode->contact_num, contact_num);
    insertnode->data = items;
    for (i = 0; i < 4; i++)
    {
        insertnode->courseID[i] = courseId[i];
    }
    insertnode->next = datanode;
    datanode = insertnode;
    fprintf(fileptr, "%i ", insertnode->data);
    fprintf(fileptr, "%s ", insertnode->name);
    fprintf(fileptr, "%s ", insertnode->address);
    fprintf(fileptr, "%s ", insertnode->email);
    fprintf(fileptr, "%s ", insertnode->contact_num);
    for (i = 0; i < 4; i++)
    {
```

```
        fprintf(fileptr, "%i ", datanode->courseID[i]);
    }
    fclose(fileptr);
}
```

Searching Code :

```
void search()
{
    printf("inside the search method");
    fetech();
    struct info *searchnode = datanode;
    int rollnumber, i;
    int found = -1;
    printf("\nEnter the roll number of student :\n");
    scanf("%i", &rollnumber);
    while (searchnode != NULL)
    {
        if (searchnode->data == rollnumber)
        {

            printf("name = %s \n", searchnode->name);
            printf("Address =%s\n", searchnode->address);
            printf("Contact number = %s\n",searchnode->contact_num);
            printf("E-mail = %s\n",searchnode->email);

            for (i = 0; i < 4; i++)
            {

                printf("cousrse id of subject %i = %i \n", i + 1, searchnode-
>courseID[i]);
            }
            printf("=============================================\n");
            found = 1;
            return;
            break;
        }

        searchnode = searchnode->next;
    }
    if (found != 1)
    {
        printf("student with roll number %i is not found !!!!\n", rollnumber);
    }
}
```

Update Code :

```
void update()
{
    printf("\ninside the update method\n");
    datanode = NULL;
    fetech();
    struct info *updatenode = datanode;
    int rollnumber;

    int found = -1;
    printf("\nEnter the roll number of student :\n");
    scanf("%i", &rollnumber);
    while (updatenode != NULL)
    {

        if (updatenode->data == rollnumber)
        {
          //entering new data
            printf("Record with roll number %d Found !!!\n", rollnumber);
            printf("\n Enter new name :");
            scanf("%s", updatenode->name);
            printf("\nEnter the address of student :\n");
            scanf("%s", updatenode->address);
            printf("\nEnter the email of student :\n");
            scanf("%s", updatenode->email);
            printf("\nEnter the contact number of student:\n");
            scanf("%s", updatenode->contact_num);
            printf("\nupdation sucessfull\n");
            found = 1;
        }
        updatenode = updatenode->next;
    }
    writeinfile("w");
    if (found == -1)
    {
        printf("\nstudent with roll number %i is not found !!!!\n", rollnumber);
    }
}
```

Delete Code :

```
void delete ()
```

```
{
    printf("\nInside the delete method :\n");
    datanode = NULL;
    fetech();
    struct info *tempdeletenode1 = datanode, *tempdeletenode2 = datanode;
    int rollnumber;
    int found = -1;
    printf("\nEnter the roll number of student :\n");
    scanf("%i", &rollnumber);
    while (tempdeletenode1 != NULL)
    {

        if (tempdeletenode1->data == rollnumber)
        {

            printf("Record with roll number %d Found !!!\n", rollnumber);

            if (tempdeletenode1 == tempdeletenode2)
            {
                // this condition will run if
                // the record that we need to delete is the first node
                // of the linked list
                datanode = datanode->next;
                free(tempdeletenode1);
            }
            else
            {
                // tempdeletenode1 is the node we need to delete
                // tempdeletenode2 is the node previous to tempdeletenode1
                tempdeletenode2->next = tempdeletenode1->next;
                free(tempdeletenode1);
            }

            found = 1;
            printf("Record Successfully Deleted !!!\n");
        }
        tempdeletenode2 = tempdeletenode1;
        tempdeletenode1 = tempdeletenode1->next;
    }
    writeinfile("w");
    if (found == -1)
        printf("Student with roll number %d is not found !!!\n", rollnumber);
}
```

Terminal display code :

```
void display()
```

```
{
    datanode = NULL;
    fetech();
    struct info *outputlink = datanode;
    printf("\nThe data inside the file :\n");
    int i = 0;
    while (outputlink != NULL)
    {
        printf("=============================================\n");
        printf("name = %s \n", outputlink->name);
        printf("rollno =%i\n", outputlink->data);
        printf("Email =%s\n",outputlink->email);
        printf("address =%s\n", outputlink->address);
        printf("contact number =%s\n", outputlink->contact_num);
        for (int i = 0; i < 4; i++)
        {

            printf("cousrse id of subject %i = %i \n", i + 1, outputlink-
>courseID[i]);
        }
        printf("=============================================\n");

        i++;
        outputlink = outputlink->next;
    }
}
```

Main function :

```c
int main()
{

    //variable for running switch
    char switch_choice;
    //variable to run the loop continously
    char lopping_option;
    //indicating the progrma is about student management system
    printf("\n\t\t\t==========STUDENT MANAGEMENT SYSTEM==========\n");
    // showing the user what kind of operation to perform

    do
    {
        getchar();

        printf("\nPress : \ni.To Insert the data .\nu. To Update the data .\nd. T
o Delete the data.\ns. To Search the data.\np. To print data on terminal\n");

        //asking user input to run switch operation

        scanf("%c", &switch_choice);
        //executing switch command
        switch ((switch_choice))
        {
        //conditions to perform the insert, update, delete, search operationy
        case 'i':
            insert();
            break;
        case 'u':
            update();
            break;
        case 'd':
            delete ();
            break;
        case 's':
            search();
            break;
        case 'p':
            display();
            break;

        default:
            printf("\n===Sorry!!! Invalid operation.===\n");
            break;
        }
```

```
        //deleting all the extra characters
        getchar();
        printf("\nDo you want to continue(y/n)?\n");
        //asking user to continue or not
        scanf("%c", &lopping_option);
    }
    //checking the condition to run the loop or not
    while (lopping_option == 'y');

    return 0;
}
```