Week-2 UE20CS207 DSLAB

- Name : P K Navin Shrinivas
- SRN : PES2UG20CS237
- Section : D
- Batch : 2

Assginment problem 1 : Contacts,store using doubly linked list and display.

Code :

main.c

```c
#include "1_1.h"
#include <stdio.h>
#include <stdlib.h>

int main(){
    struct head* top = (struct head*)malloc(sizeof(struct head));
    top->link=NULL;//init struct
    top->num = 0;
    while(true)
    {
        printf("1.Enter details\n");
        printf("2.Display details\n");
        printf("Anything else for exit.\n");
        printf("Choice : ");
        int choice;
        scanf("%d",&choice);
        if(choice == 1)
        {
            enterdetails(top);
        }
        else if(choice == 2)
        {
            display(top);
        }
    }
}
```

1_1.h

```c
#include<stdio.h>
#include <stdlib.h>
#include <stdbool.h>



struct head{
    struct details* link;
    int num;
};
struct details{
    char name[100];
    int number;
    char address[1000];
    struct details *llink,*rlink;
};
void enterdetails(struct head* top);
void display(struct head* top);
```

1_1.c

```c
#include <stdio.h>
#include <stdlib.h>
#include "1_1.h"

void enterdetails(struct head* top)
{
    if(top->link==NULL)
    {
        struct details* temp=(struct details*)malloc(sizeof(struct details));
        temp->llink=NULL;
        temp->rlink=NULL;
        top->link=temp;
        printf("Enter Details : \n");
        printf("Name : ");
        scanf("\t%[^\n]*c",temp->name);
        printf("Phone number : ");
        scanf("%i",temp->number);
        printf("Enter address : ");
        scanf("\t%[^\n]%*c",temp->address);
        top->num=top->num+1;
        return;


    }
    struct details* curr=top->link;
    while(curr->rlink!=NULL){
        curr=curr->rlink;
    }
    struct details* temp = (struct details*)malloc(sizeof(struct details));
    temp->llink=curr;
    curr->rlink=temp;
    temp->llink=NULL;//the details are always being added to the end of the linked list
    printf("Enter Details : \n");
    printf("Name : ");
    scanf("%s",temp->name);
    printf("Phone number : ");
    scanf("%d",temp->number);
    printf("Enter address : ");
    scanf("\t%[^\n]%*c",temp->address);
    top->num=top->num+1;
    return;
}

void display(struct head* top)
{
    if(top->link==NULL)
    {
        printf("Contats register is empty!\n");
        return;
    }
    struct details* curr = top->link;
    int i=1;
    while(curr!=NULL)
    {
        printf("Details of person %d ",i);
        i+=1;
        printf("Name : %s",curr->name);
        printf("Phone number : %lu",curr->number);
        printf("Address : %d",curr->address);
        curr=curr->rlink;
    }

}
```

Problem 2 : Employee program

Code :

main.c :

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

typedef struct contact
{
    char ssn[15];
    char name[20];
    char dept[20];
    char desg[30];
    float salary;
    char phone[11];
    int age;
    struct contact *rlink;
    struct contact *llink;
} node;

node *start = NULL;
void insert();
void delete_ag();
void display();
void display_sel();
void sub_dis(node *);
void main()
{
    int choice;
    do
    {
        printf("\nEnter Your choice:- \n1.Insert Record \n2.Delete Data Whose age is above n:- \n3.Display ba
        //        printf("\nEnter Your chooice:-\n1.Inset Front\n2.Insert Rear\n3.Insert Position\n4.Delete F
        scanf("%d",&choice);
        switch (choice)
        {
    case 1 : insert();   break;
    case 2 : delete_ag();break;
    case 3 : display_sel();break;
    case 4 : display();break;
    case 5 : exit(0);

        default:  printf("Enter a valid response");
            break;
        }
    } while (1);

}

void insert()
{
    node *temp=(node *)malloc(sizeof(node));
    if(temp==NULL)
        printf("\n===Memory Overflow===\n");
    else
    {
        printf("Enter SSN:- ");
        scanf("%s",temp->ssn);
        printf("Enter Name:- ");
        scanf("%s",temp->name);
        printf("Enter Dept:- ");
        scanf("%s",temp->dept);
        printf("Enter Designation:- ");
        scanf("%s",temp->desg);
        printf("Enter Salary:- ");
        scanf("%f",&(temp->salary));
        printf("Enter Phone No. :- ");
        scanf("%s",temp->phone);
        printf("Enter Age:- ");
        scanf("%d",&(temp->age));
        temp->rlink=NULL;
```

```
        temp->llink=NULL;
        if(start==NULL)
        {
            start=temp;
        }
        else
        {
            node *trav=start;
            while(trav->rlink!=NULL)
            {
                trav=trav->rlink;
            }
            temp->llink=trav;
            trav->rlink=temp;
        }
    }
}
void display()
{
    if(start==NULL)
        printf("The contact list is Empty");
    else
    {
        node *trav=start;
        while(trav!=NULL)
        {
            printf("\nSSN:- %s\nName:- %s\nDepartment:- %s\nDesignation:- %s\nSalary:- %0.2f\nPhone No.:- %s\
    trav=trav->rlink;
        }
        printf("\nEnd of the List\n\n");
    }
}
void delete_ag()
{
    int ag,found;
    printf("The age above which you want the data to be deleted:- ");
    scanf("%d",&ag);
    if(start==NULL)
        printf("The List is Empty\n");
    else
    {
        node *trav=start;
        while(trav!=NULL)
        {
            if(trav->age >= ag)
            {
                node *prev=trav->llink;
                node *temp=trav;
                if(prev!=NULL)
                    prev->rlink=trav->rlink;
                if(trav->rlink!=NULL)
                    trav->rlink->llink=prev;
                if(start==trav)
                    start=trav->rlink;
                trav=trav->rlink;
                free(temp);
            }
            else
                trav=trav->rlink;
        }
    printf("Deleted all records of age above %d",ag);
    }
}
void display_sel()
{
    char arr[20];
    printf("Enter the departments:- ");
    scanf("%s",arr);
    if(start==NULL)
        printf("The list is empty.");
    else
    {
        node *trav=start;
        int count=0;
```

```
        while(trav!=NULL)
        {
            if(strcmp(arr,trav->dept)==0)
            {
                sub_dis(trav);
                trav=trav->rlink;
                count++;
            }
            else
                trav=trav->rlink;

        }
        if(count==0)
            printf("There No Employes in %s Department\n",arr);
        else
            printf("No of Employes Displayed is %d\n",count);
    }
    printf("\nEnd of record");
}
void sub_dis(node *trav)
{
    printf("\nSSN:- %s\nName:- %s\nDepartment:- %s\nDesignation:- %s\nSalary:- %0.2f\nPhone No.:- %s\nAge:- %
}
```