```
/* 3a)  Design , develop and implement a menu driven program in C for the following
operations on SLL of integer data.
a) Create a SLL stack of N integers
b) Display of SLL
c) Linear search
d) Concatenations of two SLL   */

#include<stdio.h>
#include<stdlib.h>

struct node
{
   int data;
   struct node* link;
};

struct node * create_node()
{
   struct node* newNode;
   newNode=malloc(sizeof(struct node));
   if(newNode == NULL)
   {
      printf("No Memory allocated\n");
      return;
   }
   printf("Enter the data\n");
   scanf("%d",&newNode->data);
   newNode->link=NULL;
   return newNode;
};

struct node* insert_end(struct node * head)
{
   struct node *temp,*newNode;
   temp=head;
   newNode=create_node();
   if(head==NULL)
   {
      head=newNode;
      temp=head;
   }
   else
   {
      while(temp->link!=NULL)
         temp=temp->link;
      temp->link=newNode;
   }
```

```
    return head;
}


void display(struct node *head)
{
    struct node* temp;
    if(head==NULL)
    {
        printf("\nEmpty list\n");
        return;
    }
    temp=head;
    printf("\n The elements are : ");
    while(temp)
    {
        printf(" %d ",temp->data);
        temp=temp->link;
    }
   printf("\n");
}


void search_list(struct node* head, int key)
{
    struct node*temp=head;
    int pos=0;
    while(temp!=NULL)
    {
        pos++;
        if(key==temp->data)
        {
            printf("\nThe key element %d found at position :%d \n ",temp->data,pos);
            return;
        }
        temp=temp->link;
    }
    printf("\nThe key element %d not found in the list\n",key);
}

struct node* create_list()
{
    struct node *temp,*head=NULL,*newNode;
    int n,i;
    printf("  How many nodes? : ");
    scanf("%d", &n);
    for(i=0;i<n;i++)
```

```
      {
         newNode=create_node();
         if(head==NULL)
            head=temp=newNode;
         else
            {
               temp->link=newNode;
               temp=newNode;
            }
      }
   return head;
}

struct node* concatenate_list(struct node* head1, struct node* head2)
{
   struct node *temp;
   if(head1==NULL)
      return head2;
   if(head2==NULL)
      return head1;
   temp=head1;
   while(temp->link!=NULL)
      temp=temp->link;
   temp->link=head2;
   return head1;
}

int main()
{
   int choice,ch=1,key,n,i;
   struct node* stack=NULL, *list1,*list2,*list;
   while(1)
   {
      printf("\n");
      printf("1. Create a stack of N integers\n");
      printf("2. Display\n");
      printf("3. Linear search\n");
      printf("4. Concatenation of two SLL\n");
      printf("5. Exit\n");
      printf("Enter your choice\n");
      scanf("%d",&choice);
      switch(choice)
      {
         case 1: printf("Creating a STACK of N integers\n");
               printf("Enter the value of N\n");
               scanf("%d",&n);
               for(i=0;i<n;i++)
                  stack=insert_end(stack);
```

```
                break;
            case 2: display(stack);
                break;
            case 3: printf("Enter the key element to search: \t");
                scanf("%d",&key);
                search_list(stack,key);
                break;
            case 4: printf("\nCreate List1\n");
                list1=create_list();
                printf("\nCreate List2\n");
                list2=create_list();
                printf("\nList1:  \n");
                display(list1);
                printf("\nList2:  \n");
                display(list2);
                list=concatenate_list(list1,list2);
                printf("\n The concatenated list is :  \n");
                display(list);
                break;

            case 5: exit(1);
            default: printf("Enter the correct choice\n");
                  break;
        }
    }
    return 0;
}

/*output


1. Create a stack of N integers
2. Display
3. Linear search
4. Concatenation of two SLL
5. Exit
Enter your choice
1
Creating a STACK of N integers
Enter the value of N
5
Enter the data
2
Enter the data
9
Enter the data
5
Enter the data
```

```
4
Enter the data
1

1. Create a stack of N integers
2. Display
3. Linear search
4. Concatenation of two SLL
5. Exit
Enter your choice
2

 The elements are :  2  9  5  4  1

1. Create a stack of N integers
2. Display
3. Linear search
4. Concatenation of two SLL
5. Exit
Enter your choice
3
Enter the key element to search:        5

The key element 5 found at position :3

1. Create a stack of N integers
2. Display
3. Linear search
4. Concatenation of two SLL
5. Exit
Enter your choice
3
Enter the key element to search:        10

The key element 10 not found in the list

1. Create a stack of N integers
2. Display
3. Linear search
4. Concatenation of two SLL
5. Exit
Enter your choice
4

Create List1
  How many nodes? : 3
Enter the data
2
```

```
Enter the data
4
Enter the data
1

Create List2
  How many nodes? : 2
Enter the data
7
Enter the data
5

List1:

 The elements are :  2  4  1

List2:

 The elements are :  7  5

 The concatenated list is :

 The elements are :  2  4  1  7  5

1. Create a stack of N integers
2. Display
3. Linear search
4. Concatenation of two SLL
5. Exit
Enter your choice

*/
```

/* 3b. Create a SLL Queue of N Students Data. */

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

struct node
{

   int sem,phno;
   char name[20],branch[20],usn[20];
   struct node *link;

}*head=NULL,*newnode;
```

```
void create()
{
   int sem,phno;
   char name[20],branch[20],usn[20];
   newnode=(struct node*)malloc(sizeof(struct node));
   printf("Enter the student details\n");
   printf("USN: \n");
   scanf("%s",&usn);
   printf("Name: \n");
   scanf("%s",&name);
   printf("Branch: \n");
   scanf("%s",&branch);
   printf("Sem: \n");
   scanf("%d",&sem);
   printf("Phone Number: \n");
   scanf("%d",&phno);
   strcpy(newnode->usn,usn);
   strcpy(newnode->name,name);
   strcpy(newnode->branch,branch);
   newnode->sem=sem;
   newnode->phno=phno;
   newnode->link=NULL;
}

void insert_end()
   {
      struct node *temp;
      if(head==NULL)
         {
            create();
            head=newnode;
            temp=head;
         }
      else
         {
            create();
            temp->link=newnode;
            temp=newnode;
         }

   }

void display()
   {

      struct node* temp=head;
      if(head==NULL)
```

```
         {
            printf("\n Queue is empty");
            return;
         }
      printf("The Student details: \n");
      while(temp!=NULL)
      {
      printf("USN:%s\nName: %s\nBranch: %s\nSem: %d\nPhone Number: %d\n\n",temp->usn,temp-
>name,temp->branch,temp->sem,temp->phno);
      temp=temp->link;
      }
   }
void delete_front()
   {

      struct node *temp;
      temp=head;
      if(temp->link==NULL)
         {
            free(temp);
            head=NULL;
            return 0;
         }
      else
         {
            head=temp->link;
            printf("USN:%s\nName: %s\nBranch: %s\nSem: %d\nPhone Number: %d\n",temp->usn,temp-
>name,temp->branch,temp->sem,temp->phno);
            free(temp);
         }
      return 0;

   }

int main()
{
   int ch=1,n,i;
   while(ch)
   {
      printf("1. Create a SLL Queue of N Students data\n");
      printf("2. Delete Queue\n");
      printf("3. Display\n");
      printf("4. Quit\n");
      printf("enter your choice\n");
      scanf("%d",&ch);
      switch(ch)
      {
         case 1: printf("Creating Queue of N student list\n");
```

```
          printf("\nEnter the value of N \n");
          scanf("%d",&n);
          for(i=0;i<n;i++)
            insert_end();
          break;
      case 2: delete_front();
          break;
      case 3: display();
          break;
      case 4: exit(1);
      default: printf("Wrong choice\n");
            break;
    }
  }
  return 0;
}

/* Output:

1. Create a SLL Queue of N Students data
2. Delete Queue
3. Display
4. Quit
enter your choice
1
Creating Queue of N student list

Enter the value of N
3
Enter the student details
USN:
12
Name:
abc
Branch:
cs
Sem:
3
Phone Number:
75342234
Enter the student details
USN:
32
Name:
xyz
Branch:
aiml
Sem:
```

```
4
Phone Number:
32453453
Enter the student details
USN:
45
Name:
qwert
Branch:
aiml
Sem:
3
Phone Number:
987654546
1. Create a SLL Queue of N Students data
2. Delete Queue
3. Display
4. Quit
enter your choice
3
The Student details:
USN:12
Name: abc
Branch: cs
Sem: 3
Phone Number: 75342234

USN:32
Name: xyz
Branch: aiml
Sem: 4
Phone Number: 32453453

USN:45
Name: qwert
Branch: aiml
Sem: 3
Phone Number: 987654546

1. Create a SLL Queue of N Students data
2. Delete Queue
3. Display
4. Quit
enter your choice
2
USN:12
Name: abc
Branch: cs
```

```
Sem: 3
Phone Number: 75342234
1. Create a SLL Queue of N Students data
2. Delete Queue
3. Display
4. Quit
enter your choice
3
The Student details:
USN:32
Name: xyz
Branch: aiml
Sem: 4
Phone Number: 32453453

USN:45
Name: qwert
Branch: aiml
Sem: 3
Phone Number: 987654546

1. Create a SLL Queue of N Students data
2. Delete Queue
3. Display
4. Quit
enter your choice
*/
```