# SSN College of Engineering, Kalavakkam

# Department of Computer Science and Engineering

# III Semester - CSE 'A ',’B’ & ‘C’

# UCS 1312 Data Structures Lab Laboratory

# Academic Year: 2019-2020 Batch: 2018-2022

# Exercise 4: Implementation of doubly linked list ADT

**Name:Rahul Ram M**

**Class:CSE - B**

**Register Number:185001121**

**Create a doubly linked list to store set of student names**

**Perform the following operations using a menu driven program**

**1. Insert student name in the front of the list**

**2. Insert student name at the end of the list**

**3. Insert a record after a given name in the list**

**4. Search a given student in the list**

**5. Delete a given student**

**6. Display all student names**

**7. Display the students in alphabetical order**

**PROGRAM:**

#include<stdio.h>

#include<stdlib.h>

#include<string.h>

#include"functions.h"

#include"prototype.h"

int main()

{

int s,ch;

node \*head,\*tail;

head=(node \*)malloc(sizeof(node));

tail=(node \*)malloc(sizeof(node));

head->next=tail;

head->prev=NULL;

tail->next=NULL;

tail->prev=head;

printf("Press\n1.Insert First.\n2.Insert Last.\n3.Insert after a name.\n4.Search a student's name.\n5.Delete a student name.\n6.Display all name.\n7.Display all names in alphabetical order.\n8.Exit.\n");

scanf("%d",&ch);

while(ch!=8)

{

switch(ch)

{

case 1:insertfirst(head);printf("Inserted Successfully!\n");break;

case 2:insertlast(head);printf("Inserted Successfully!\n");break;

case 3:insertmiddle(head);break;

case 4:printf("Search a name!\n");s=search(head);

if(s!=-1)

{

printf("Name found in %dth position.\n",s);

}

break;

case 5:delete(head);break;

case 6:display(head);break;

case 7:displaya(head);display(head);break;

case 8:return 0;

default:printf("Invalid Input!\n");

}

printf("Press\n1.Insert First.\n2.Insert Last.\n3.Insert after a name.\n4.Search a student's name.\n5.Delete a student name.\n6.Display all name.\n7.Display all names in alphabetical order.\n8.Exit.\n");

scanf("%d",&ch);

}

return 0;

}

**//prototype.h**

void insertfirst(node \*hd);

void insertlast(node \*hd);

void display(node \*hd);

int search(node \*hd);

void insertmiddle(node \*hd);

void delete(node \*hd);

void displaya(node \*hd);

**//functions.h**

typedef struct studentlist

{

char name[20];

struct studentlist \*prev,\*next;

}node;

void insertfirst(node \*hd)

{

node \*temp;

temp=(node \*)malloc(sizeof(node));

printf("Enter the student's name:");

scanf("%s",temp->name);

temp->next=hd->next;

temp->prev=hd;

hd->next->prev=temp;

hd->next=temp;

}

void insertlast(node \*hd)

{

node \*p,\*tl,\*temp;

temp=(node \*)malloc(sizeof(node));

for(p=hd->next;p->next!=NULL;p=p->next)

{

tl=p->next;

}

printf("Enter the student's name:");

scanf("%s",temp->name);

temp->next=tl;

temp->prev=tl->prev;

tl->prev->next=temp;

tl->prev=temp;

}

void insertmiddle(node \*hd)

{

node \*mid,\*temp;

int s;

s=search(hd);

if(s==-1)

{

return;

}

mid=hd->next;

for(int i=0;i<s;i++)

{

mid=mid->next;

}

temp=(node \*)malloc(sizeof(node));

printf("Enter the student's name:");

scanf("%s",temp->name);

temp->next=mid;

temp->prev=mid->prev;

mid->prev->next=temp;

mid->prev=temp;

printf("Inserted Successfully!\n");

}

void delete(node \*hd)

{

printf("Enter the student's name you want to delete!\n");

int d=search(hd);

if(d==-1)

{

return;

}

node \*del,\*temp;

del=hd->next;

for(int i=0;i<(d-1);i++)

{

del=del->next;

}

temp=del;

del->next->prev=del->prev;

del->prev->next=del->next;

free(temp);

printf("Deleted Successfully!\n");

}

int search(node \*hd)

{

node \*s;

int count=0,c=0;

char sname[20];

printf("Enter the name:");

scanf("%s",sname);

for(s=hd->next;s!=NULL;s=s->next)

{

c++;

}

for(s=hd->next;s->next!=NULL;s=s->next)

{

if(strcmp(s->name,sname)==0)

{

break;

}

else

{

count++;

}

}

if((count+1)==c)

{

printf("Name not FOUND!\n");

return -1;

}

else

{

return count+1;

}

}

void display(node \*hd)

{

node \*p;

for(p=hd->next;p->next!=NULL;p=p->next)

{

printf("%s",p->name);

printf("\n");

}

}

void displaya(node \*hd)

{

node \*p,\*q;

int c=0;

char n[20];

for(p=hd->next;p!=NULL;p=p->next)

{

c++;

}

for(int i=0;i<(c-1);i++)

{

for(p=hd->next;p->next->next!=NULL;p=p->next)

{

for(q=p->next;q->next!=NULL;q=q->next)

{

if(strcmp(p->name,q->name)>0)

{

strcpy(n,p->name);

strcpy(p->name,q->name);

strcpy(q->name,n);

}

}

}

}

printf("Displaying the names in the alphabetical order!\n");

}

**OUTPUT:**

Press

1.Insert First.

2.Insert Last.

3.Insert after a name.

4.Search a student's name.

5.Delete a student name.

6.Display all name.

7.Display all names in alphabetical order.

8.Exit.

1

Enter the student's name:Rahul

Inserted Successfully!

Press

1.Insert First.

2.Insert Last.

3.Insert after a name.

4.Search a student's name.

5.Delete a student name.

6.Display all name.

7.Display all names in alphabetical order.

8.Exit.

2

Enter the student's name:Prithvi

Inserted Successfully!

Press

1.Insert First.

2.Insert Last.

3.Insert after a name.

4.Search a student's name.

5.Delete a student name.

6.Display all name.

7.Display all names in alphabetical order.

8.Exit.

3

Enter the name:Rahul

Enter the student's name:Krishna

Inserted Successfully!

Press

1.Insert First.

2.Insert Last.

3.Insert after a name.

4.Search a student's name.

5.Delete a student name.

6.Display all name.

7.Display all names in alphabetical order.

8.Exit.

6

Rahul

Krishna

Prithvi

Press

1.Insert First.

2.Insert Last.

3.Insert after a name.

4.Search a student's name.

5.Delete a student name.

6.Display all name.

7.Display all names in alphabetical order.

8.Exit.

4

Search a name!

Enter the name:Prithvi

Name found in 3th position.

Press

1.Insert First.

2.Insert Last.

3.Insert after a name.

4.Search a student's name.

5.Delete a student name.

6.Display all name.

7.Display all names in alphabetical order.

8.Exit.

5

Enter the student's name you want to delete!

Enter the name:Krishna

Deleted Successfully!

Press

1.Insert First.

2.Insert Last.

3.Insert after a name.

4.Search a student's name.

5.Delete a student name.

6.Display all name.

7.Display all names in alphabetical order.

8.Exit.

6

Rahul

Prithvi

Press

1.Insert First.

2.Insert Last.

3.Insert after a name.

4.Search a student's name.

5.Delete a student name.

6.Display all name.

7.Display all names in alphabetical order.

8.Exit.

2

Enter the student's name:Krishna

Inserted Successfully!

Press

1.Insert First.

2.Insert Last.

3.Insert after a name.

4.Search a student's name.

5.Delete a student name.

6.Display all name.

7.Display all names in alphabetical order.

8.Exit.

3

Enter the name:Krishna

Enter the student's name:

Ram

Inserted Successfully!

Press

1.Insert First.

2.Insert Last.

3.Insert after a name.

4.Search a student's name.

5.Delete a student name.

6.Display all name.

7.Display all names in alphabetical order.

8.Exit.

1

Enter the student's name:Prakash

Inserted Successfully!

Press

1.Insert First.

2.Insert Last.

3.Insert after a name.

4.Search a student's name.

5.Delete a student name.

6.Display all name.

7.Display all names in alphabetical order.

8.Exit.

6

Prakash

Rahul

Prithvi

Krishna

Ram

Press

1.Insert First.

2.Insert Last.

3.Insert after a name.

4.Search a student's name.

5.Delete a student name.

6.Display all name.

7.Display all names in alphabetical order.

8.Exit.

7

Displaying the names in the alphabetical order!

Krishna

Prakash

Prithvi

Rahul

Ram

Press

1.Insert First.

2.Insert Last.

3.Insert after a name.

4.Search a student's name.

5.Delete a student name.

6.Display all name.

7.Display all names in alphabetical order.

8.Exit.

8