USN:1BM19CS216
Name:Yashaswini Shah
Date: 23/11/2020

WAP to Implement Singly Linked List with following operations

a) Create a linked list.
b) Insertion of a node at first position, at any position and at end of list.
c) Deletion of first element, specified element and last element in the list.
d) Display the contents of the linked list

```
#include<stdio.h>
#include<stdlib.h>
struct node
{
int info;
struct node *link;
};
typedef struct node *NODE;
NODE getnode()
{
NODE x;
x=(NODE)malloc(sizeof(struct node));
if(x==NULL)
{
printf("memory full\n");
exit(0);
}
return x;
}
void freenode(NODE x)
{
free(x);
}
NODE insert_front(NODE first,int item)
{
NODE temp;
temp=getnode();
temp->info=item;
temp->link=NULL;
if(first==NULL)
return temp;
temp->link=first;
```

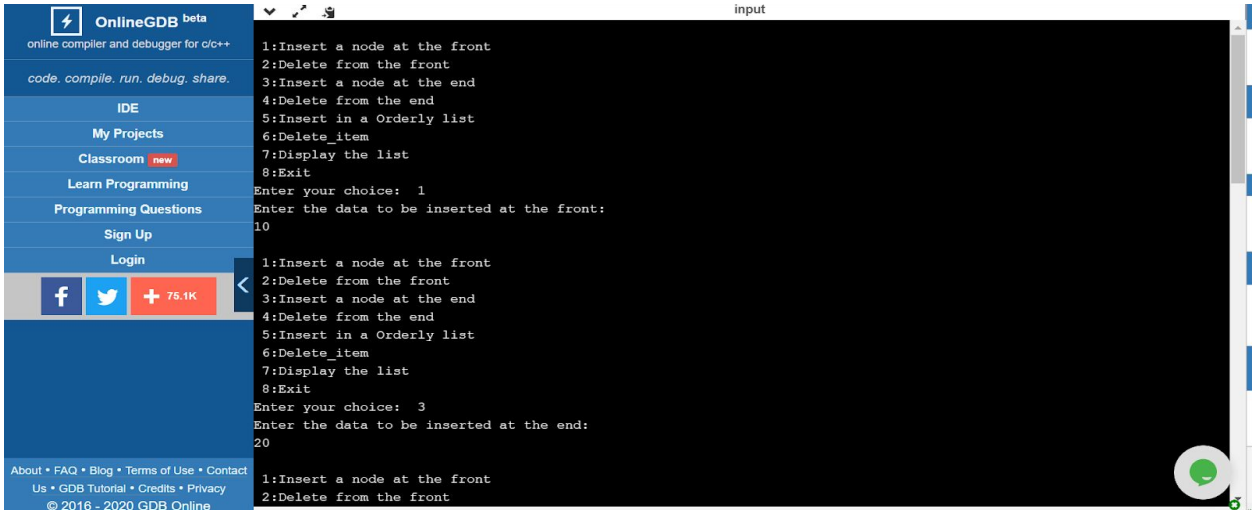
```
first=temp;
return first;
}
NODE delete_front(NODE first)
{
NODE temp;
if(first==NULL)
{
printf("list is empty cannot delete\n");
return first;
}
temp=first;
temp=temp->link;
printf("Item deleted at front-end is=%d\n",first->info);
free(first);
return temp;
}
NODE insert_rear(NODE first,int item)
{
NODE temp,cur;
temp=getnode();
temp->info=item;
temp->link=NULL;
if(first==NULL)
return temp;
cur=first;
while(cur->link!=NULL)
cur=cur->link;
cur->link=temp;
return first;
}
NODE delete_rear(NODE first)
{
NODE cur,prev;
if(first==NULL)
{
printf("list is empty cannot delete\n");
return first;
}
if(first->link==NULL)
{
printf("Item deleted is  %d\n",first->info);
free(first);
return NULL;
```

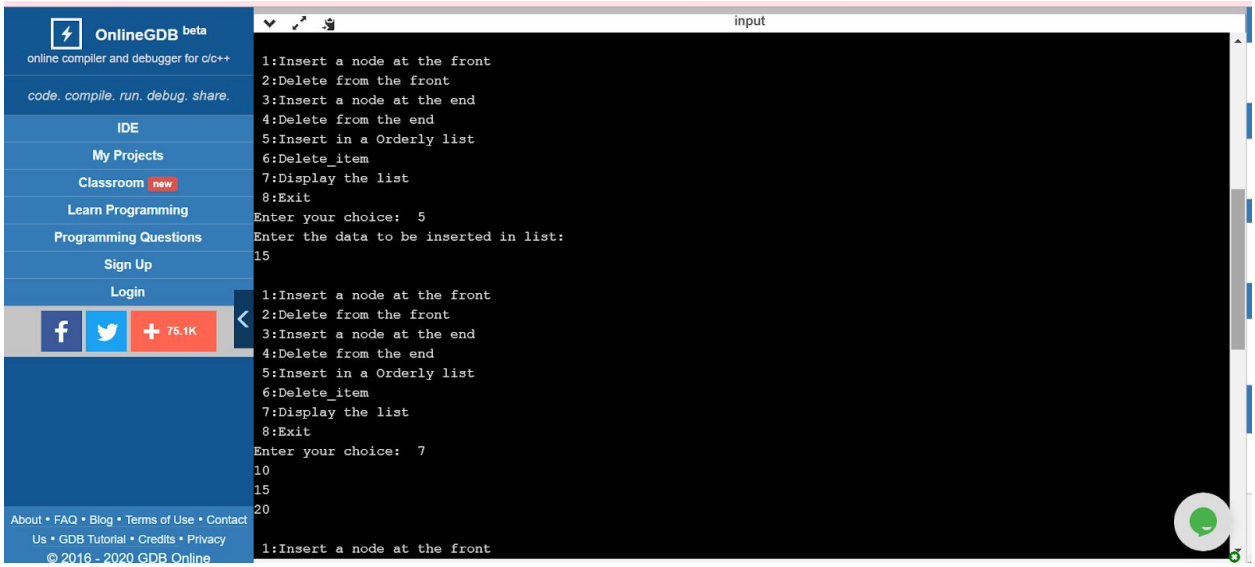
```
}
prev=NULL;
cur=first;
while(cur->link!=NULL)
{
prev=cur;
cur=cur->link;
}
printf("Item deleted at rear-end is %d",cur->info);
free(cur);
prev->link=NULL;
return first;
}
NODE order_list(int item,NODE first)
{
NODE temp,prev,cur;
temp=getnode();
temp->info=item;
temp->link=NULL;
if(first==NULL)
   return temp;
if(item<first->info)
{
temp->link=first;
return temp;
}
prev=NULL;
cur=first;
while(cur!=NULL&&item>cur->info)
{
prev=cur;
cur=cur->link;
}
prev->link=temp;
temp->link=cur;
return first;
}
NODE delete_info(int key,NODE first)
{
NODE prev,cur;
if(first==NULL)
{
printf("The list is empty\n");
return NULL;
```

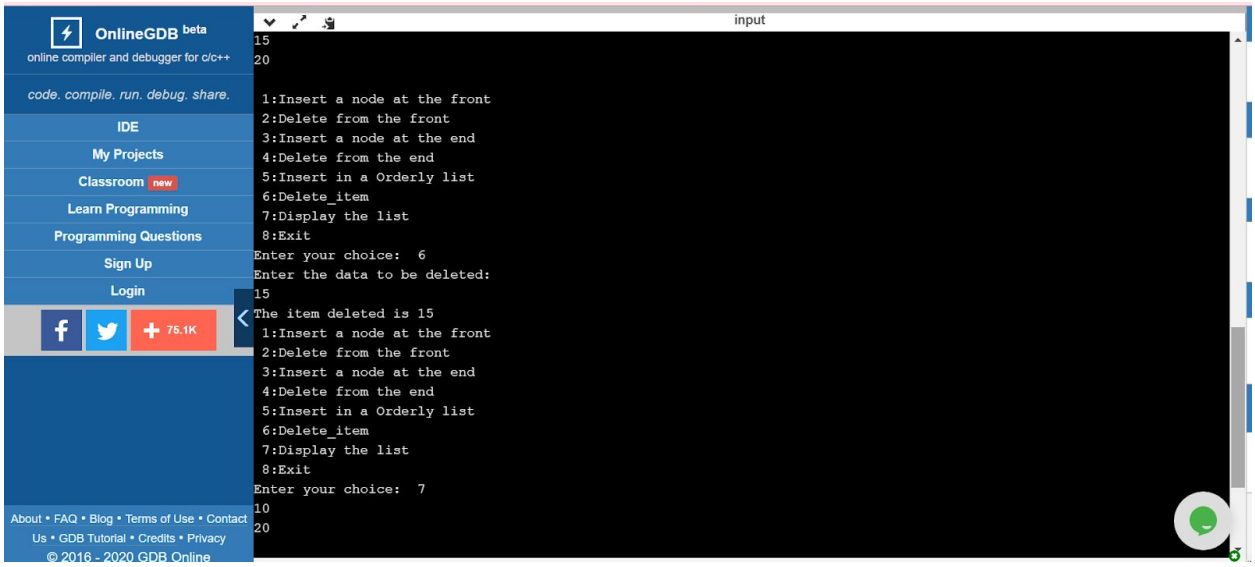
```
}
if(key==first->info)
{
cur=first;
first=first->link;
freenode(cur);
return first;
}
prev=NULL;
cur=first;
while(cur!=NULL)
{
if(key==cur->info)break;
prev=cur;
cur=cur->link;
}
if(cur==NULL)
{
printf("Search is unsuccessfull\n");
return first;
}
prev->link=cur->link;
printf("The item deleted is %d",cur->info);
freenode(cur);
return first;
}
void display(NODE first)
{
NODE temp;
if(first==NULL)
printf("The list is empty can not display items.\n");
for(temp=first;temp!=NULL;temp=temp->link)
{
printf("%d\n",temp->info);
}
}
int main()
{
int item,choice,key;
NODE first=NULL;
for(;;)
{
printf("\n 1:Insert a node at the front \n 2:Delete from the front\n 3:Insert a node at the end\n
4:Delete from the end \n 5:Insert in a Orderly list\n 6:Delete_item\n 7:Display the list\n 8:Exit\n");
```

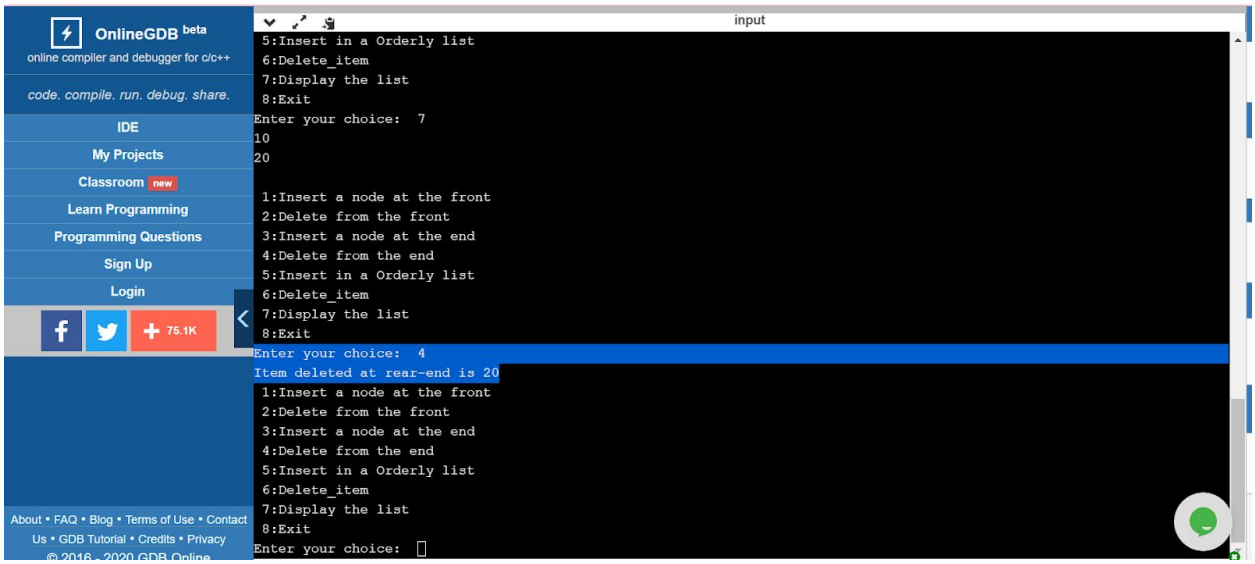
```
printf("Enter your choice:  ");
scanf("%d",&choice);
switch(choice)
{
case 1:printf("Enter the data to be inserted at the front: \n");
scanf("%d",&item);
first=insert_front(first,item);
break;
case 2:first=delete_front(first);
break;
case 3:printf("Enter the data to be inserted at the end: \n");
scanf("%d",&item);
first=insert_rear(first,item);
break;
case 4:first=delete_rear(first);
break;
case 5:printf("Enter the data to be inserted in list: \n");
scanf("%d",&item);
first=order_list(item,first);
break;
case 6:printf("Enter the data to be deleted: \n");
scanf("%d",&key);
first=delete_info(key,first);
break;
case 7:display(first);
break;
default:exit(0);
break;
}
}

}
```

OUTPUT:


OnlineGDB beta
online compiler and debugger for c/c++
code. compile. run. debug. share.
IDE
My Projects
Classroom new
Learn Programming
Programming Questions
Sign Up
Login
75.1K
About • FAQ • Blog • Terms of Use • Contact Us • GDB Tutorial • Credits • Privacy
© 2016 - 2020 GDB Online
input
1:Insert a node at the front
2:Delete from the front
3:Insert a node at the end
4:Delete from the end
5:Insert in a Orderly list
6:Delete_item
7:Display the list
8:Exit
Enter your choice: 1
Enter the data to be inserted at the front:
10
1:Insert a node at the front
2:Delete from the front
3:Insert a node at the end
4:Delete from the end
5:Insert in a Orderly list
6:Delete_item
7:Display the list
8:Exit
Enter your choice: 3
Enter the data to be inserted at the end:
20
1:Insert a node at the front
2:Delete from the front


OnlineGDB beta
online compiler and debugger for c/c++
code. compile. run. debug. share.
IDE
My Projects
Classroom new
Learn Programming
Programming Questions
Sign Up
Login
75.1K
About • FAQ • Blog • Terms of Use • Contact Us • GDB Tutorial • Credits • Privacy
© 2016 - 2020 GDB Online
input
1:Insert a node at the front
2:Delete from the front
3:Insert a node at the end
4:Delete from the end
5:Insert in a Orderly list
6:Delete_item
7:Display the list
8:Exit
Enter your choice: 5
Enter the data to be inserted in list:
15
1:Insert a node at the front
2:Delete from the front
3:Insert a node at the end
4:Delete from the end
5:Insert in a Orderly list
6:Delete_item
7:Display the list
8:Exit
Enter your choice: 7
10
15
20
1:Insert a node at the front


OnlineGDB beta
online compiler and debugger for c/c++
code. compile. run. debug. share.
IDE
My Projects
Classroom new
Learn Programming
Programming Questions
Sign Up
Login
75.1K
About • FAQ • Blog • Terms of Use • Contact Us • GDB Tutorial • Credits • Privacy
© 2016 - 2020 GDB Online
input
15
20
1:Insert a node at the front
2:Delete from the front
3:Insert a node at the end
4:Delete from the end
5:Insert in a Orderly list
6:Delete_item
7:Display the list
8:Exit
Enter your choice: 6
Enter the data to be deleted:
15
The item deleted is 15
1:Insert a node at the front
2:Delete from the front
3:Insert a node at the end
4:Delete from the end
5:Insert in a Orderly list
6:Delete_item
7:Display the list
8:Exit
Enter your choice: 7
10
20


OnlineGDB beta
online compiler and debugger for c/c++
code. compile. run. debug. share.
IDE
My Projects
Classroom new
Learn Programming
Programming Questions
Sign Up
Login
75.1K
About • FAQ • Blog • Terms of Use • Contact Us • GDB Tutorial • Credits • Privacy
© 2016 - 2020 GDB Online
input
5:Insert in a Orderly list
6:Delete_item
7:Display the list
8:Exit
Enter your choice: 7
10
20
1:Insert a node at the front
2:Delete from the front
3:Insert a node at the end
4:Delete from the end
5:Insert in a Orderly list
6:Delete_item
7:Display the list
8:Exit
Enter your choice: 4
Item deleted at rear-end is 20
1:Insert a node at the front
2:Delete from the front
3:Insert a node at the end
4:Delete from the end
5:Insert in a Orderly list
6:Delete_item
7:Display the list
8:Exit
Enter your choice: