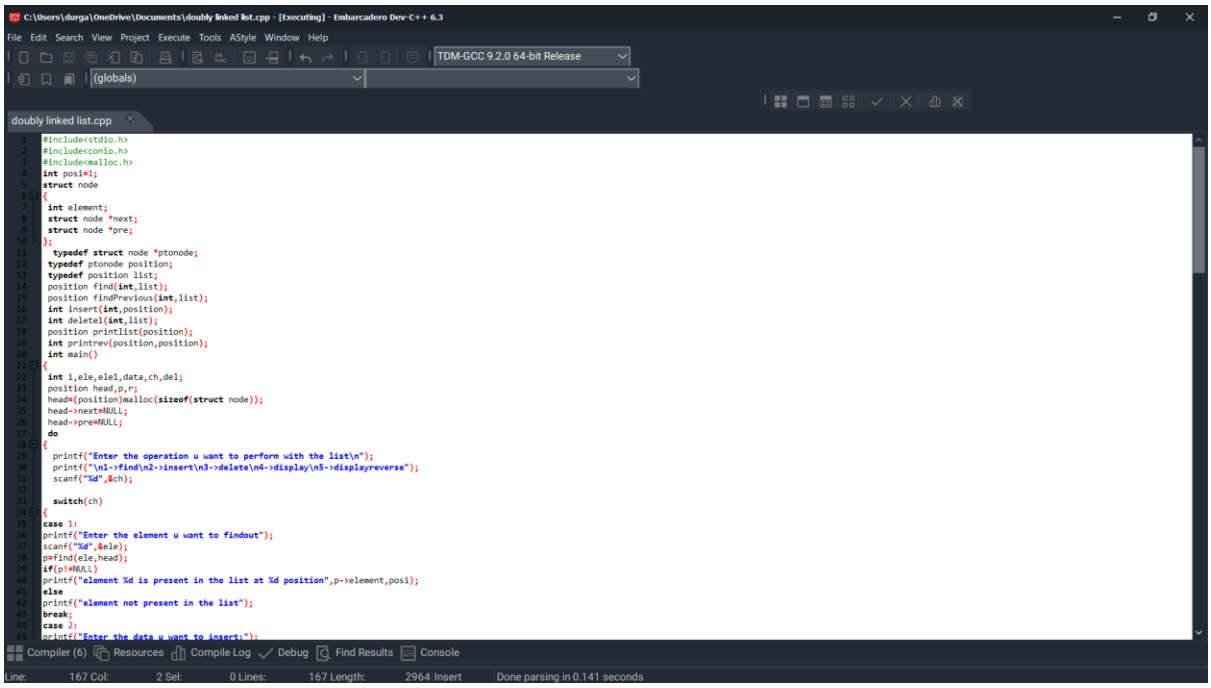
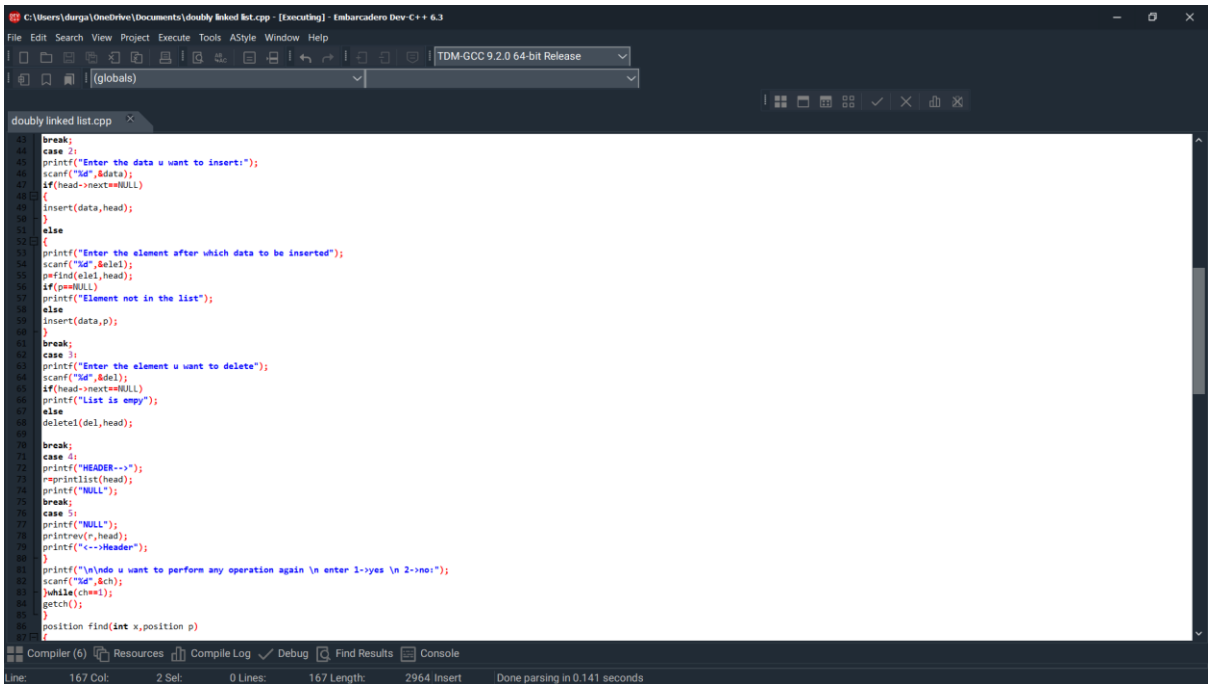
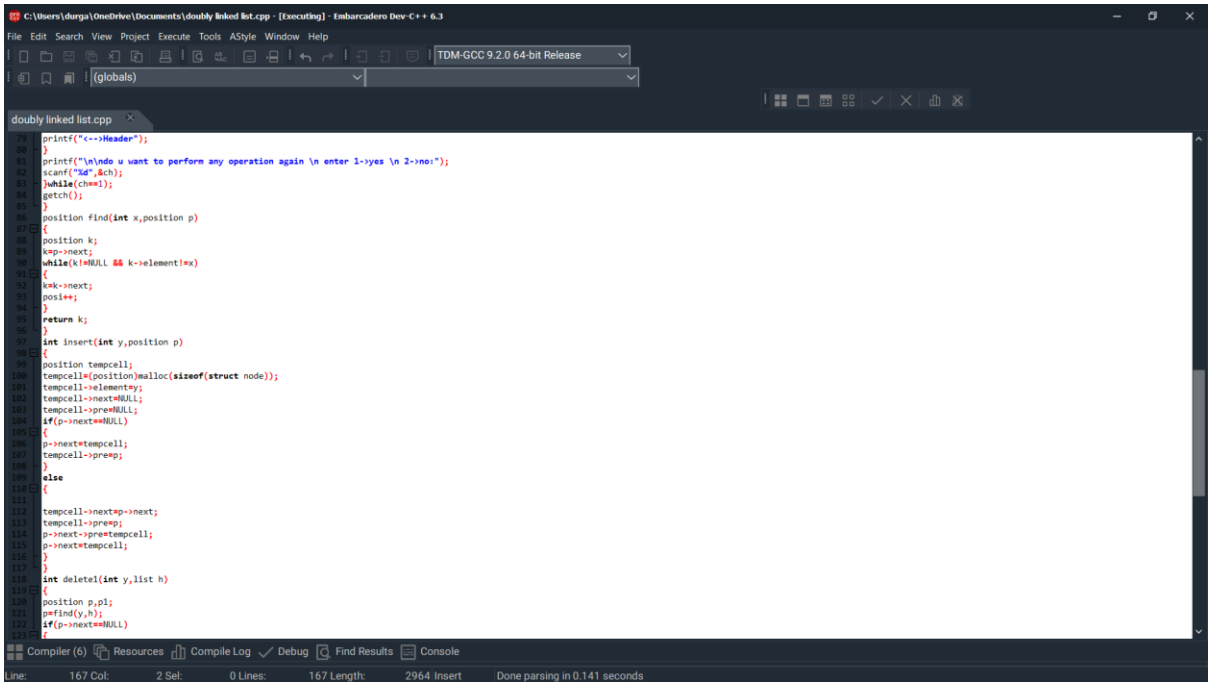
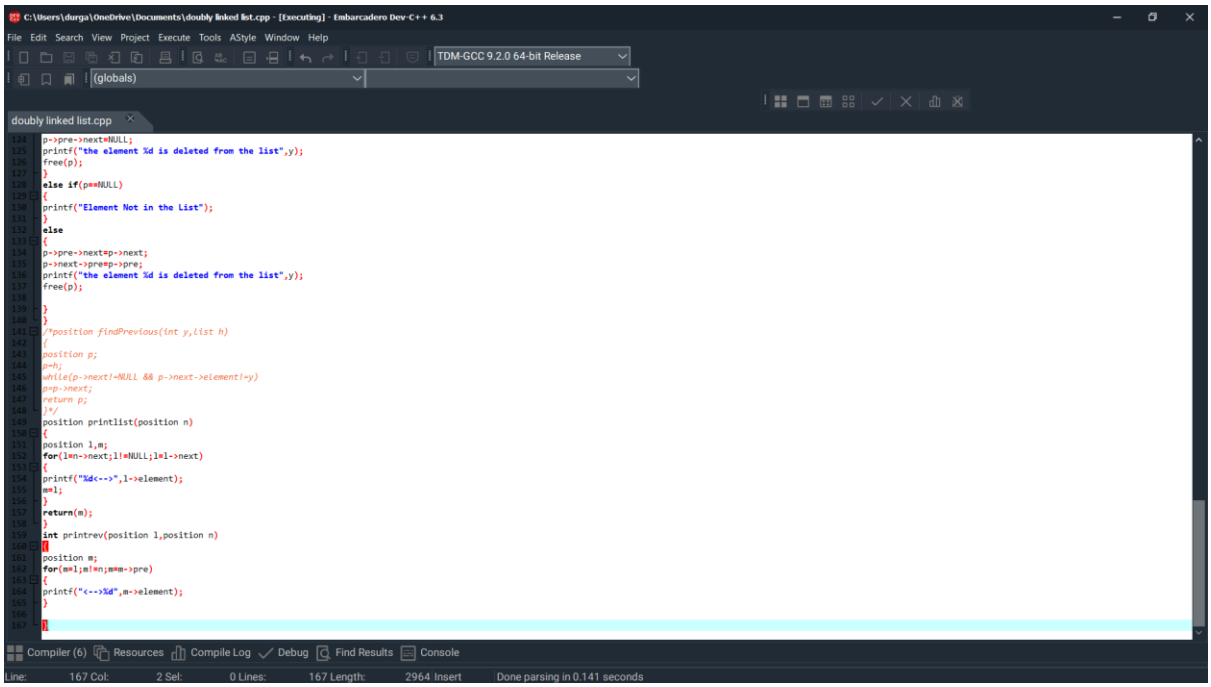
Day 4

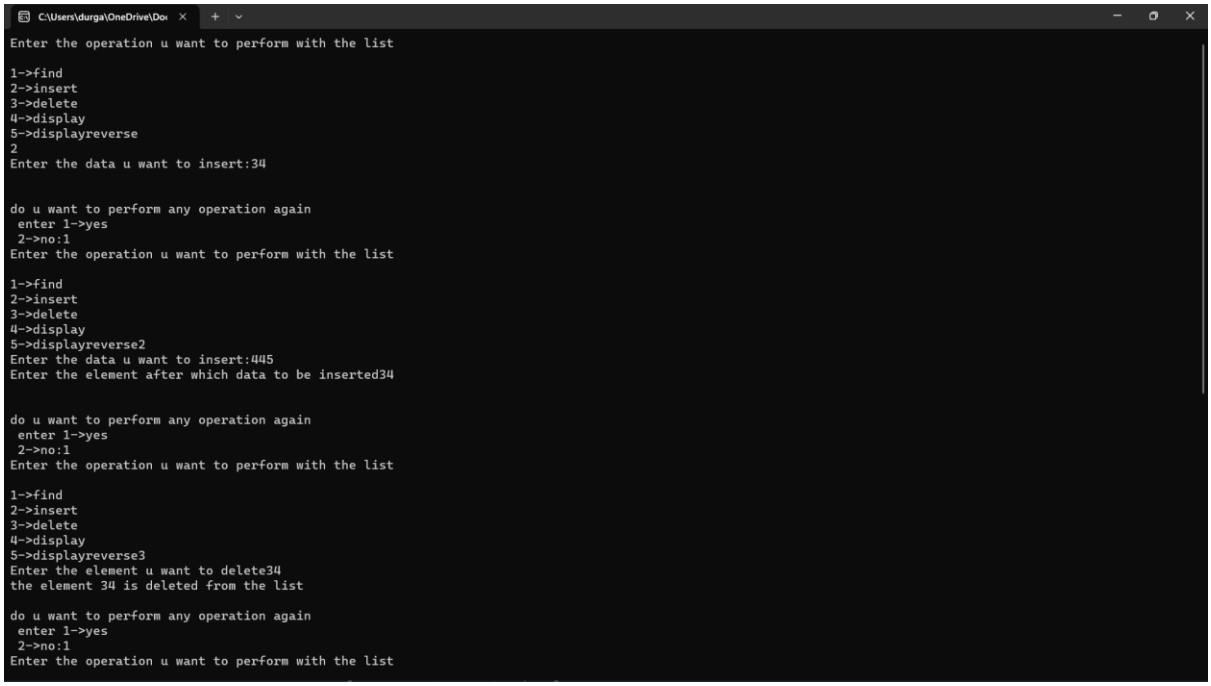
1)doubly linked list



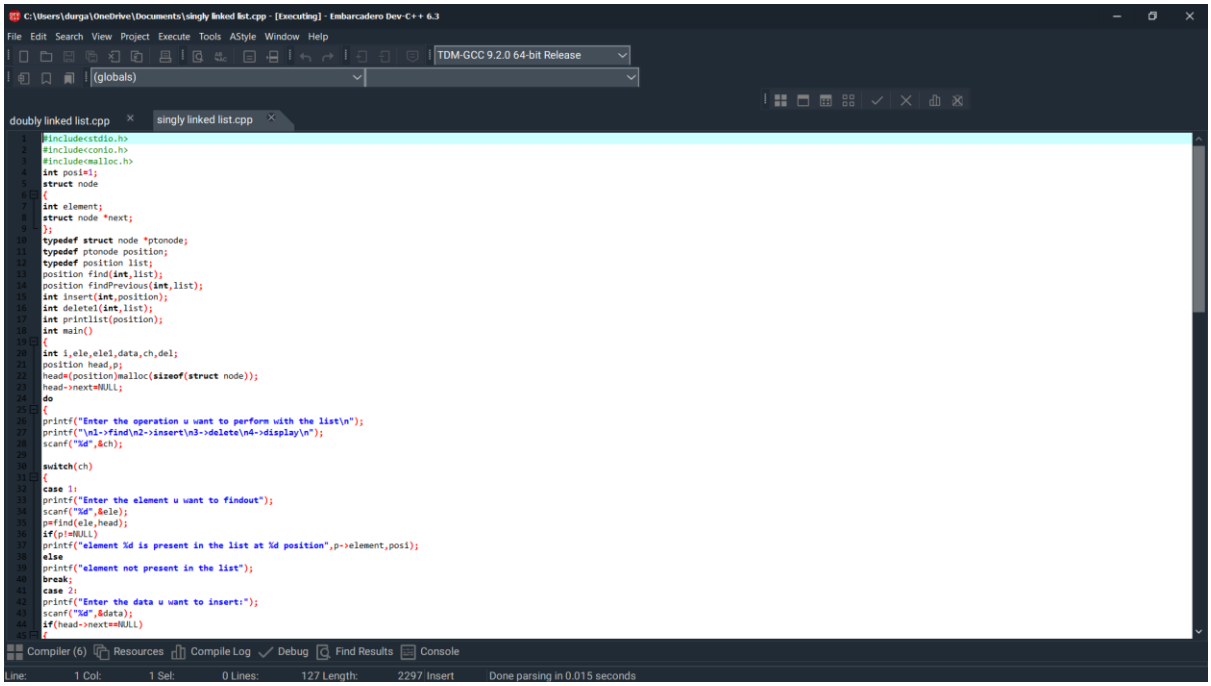


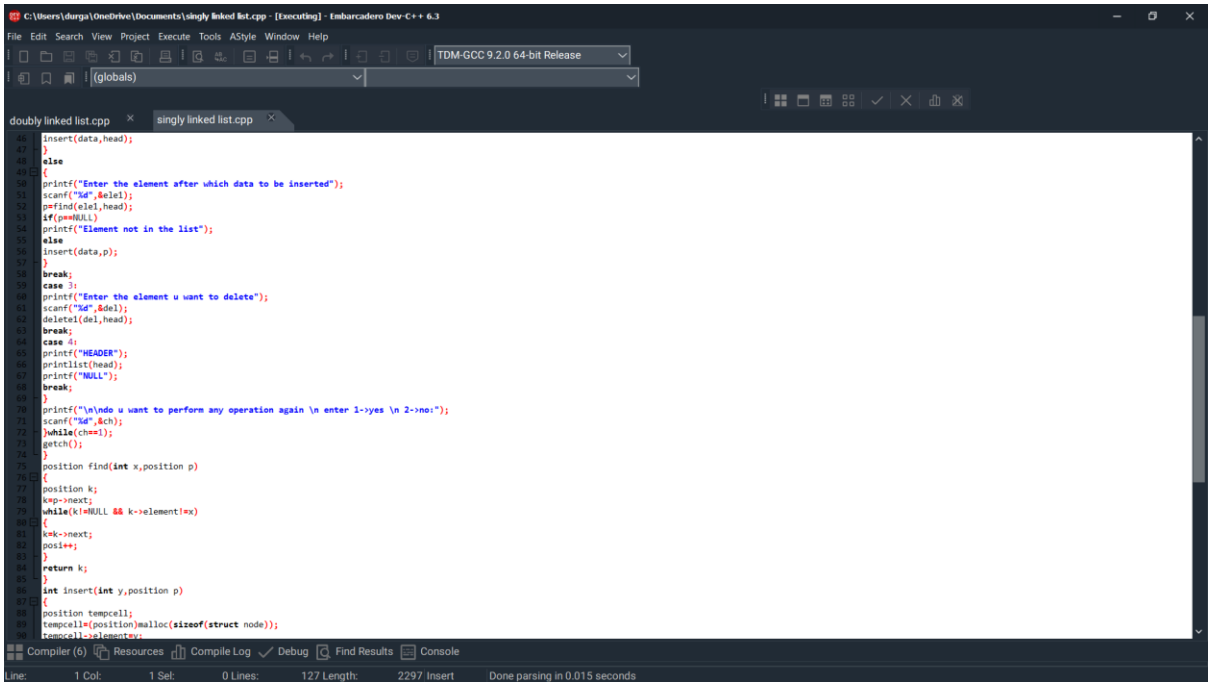

C:\Users\durga\OneDrive\Documents\doubly linked list.cpp - [Executing] - Embarcadero Dev-C++ 6.3
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 9.2.0 64-bit Release
(globals)
doubly linked list.cpp
printf("<-->Header");
}
printf("\n\ndo u want to perform any operation again \n enter 1->yes \n 2->no:");
scanf("%d",&ch);
}while(ch==1);
getch();
}
position find(int x,position p)
{
position k;
k=p->next;
while(k!=NULL && k->element!=x)
{
k=k->next;
posi++;
}
return k;
}
int insert(int y,position p)
{
position tempcell;
tempcell=(position)malloc(sizeof(struct node));
tempcell->element=y;
tempcell->next=NULL;
tempcell->pre=NULL;
if(p->next==NULL)
{
p->next=tempcell;
tempcell->pre=p;
}
else
{
tempcell->next=p->next;
tempcell->pre=p;
p->next->pre=tempcell;
p->next=tempcell;
}
}
int delete1(int y,list h)
{
position p,p1;
p=find(y,h);
if(p->next==NULL)
{
Compiler (6) Resources Compile Log Debug Find Results Console
Line: 167 Col: 2 Sel: 0 Lines: 167 Length: 2964 Insert Done parsing in 0.141 seconds

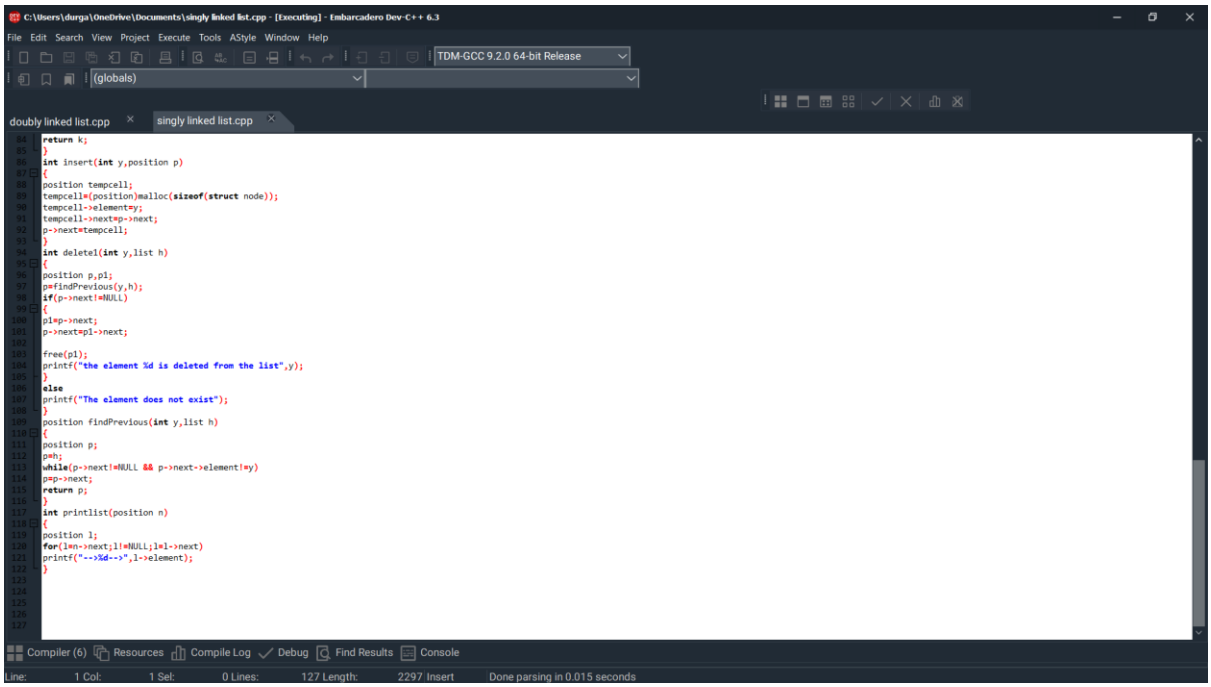

C:\Users\durga\OneDrive\Documents\doubly linked list.cpp - [Executing] - Embarcadero Dev-C++ 6.3
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 9.2.0 64-bit Release
(globals)
doubly linked list.cpp
p->pre->next=NULL;
printf("the element %d is deleted from the list",y);
free(p);
}
else if(p==NULL)
{
printf("Element Not in the List");
}
else
{
p->pre->next=p->next;
p->next->pre=p->pre;
printf("the element %d is deleted from the list",y);
free(p);
}
}
/*position findPrevious(int y,list h)
{
position p;
p=h;
while(p->next!=NULL && p->next->element!=y)
p=p->next;
return p;
}*/
position printlist(position n)
{
position l,m;
for(l=n->next;l!=NULL;l=l->next)
{
printf("%d<-->",l->element);
m=l;
}
return(m);
}
int printrev(position l,position n)
{
position m;
for(m=l;m!=n;m=m->pre)
{
printf("<-->%d",m->element);
}
}
Compiler (6) Resources Compile Log Debug Find Results Console
Line: 167 Col: 2 Sel: 0 Lines: 167 Length: 2964 Insert Done parsing in 0.141 seconds

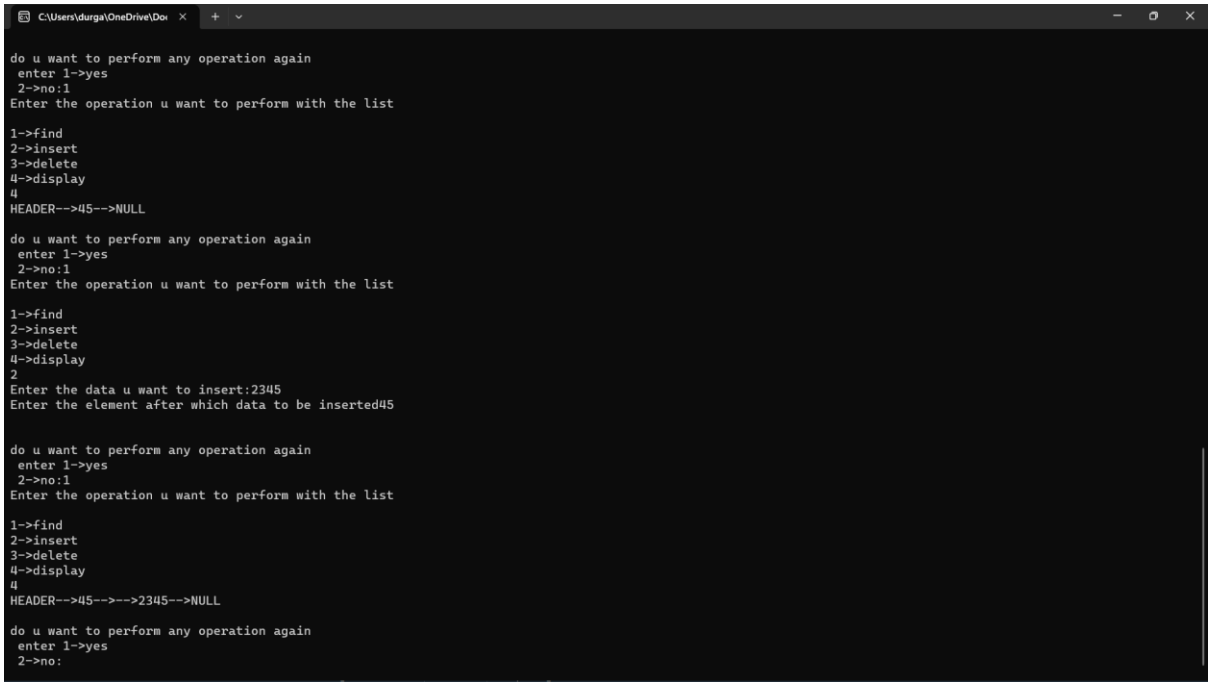


2)singly linked list




C:\Users\durga\OneDrive\Documents\singly linked list.cpp - [Executing] - Embarcadero Dev-C++ 6.3
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 9.2.0 64-bit Release
(globals)
doubly linked list.cpp
singly linked list.cpp
insert(data,head);
}
else
{
printf("Enter the element after which data to be inserted");
scanf("%d",&elel);
p=find(elel,head);
if(p==NULL)
printf("Element not in the list");
else
insert(data,p);
}
break;
case 3:
printf("Enter the element u want to delete");
scanf("%d",&del);
deletel(del,head);
break;
case 4:
printf("HEADER");
printlist(head);
printf("NULL");
break;
}
printf("\n\ndo u want to perform any operation again \n enter 1->yes \n 2->no:");
scanf("%d",&ch);
}while(ch==1);
getch();
}
position find(int x,position p)
{
position k;
k=p->next;
while(k!=NULL && k->element!=x)
{
k=k->next;
posi++;
}
return k;
}
int insert(int y,position p)
{
position tempcell;
tempcell=(position)malloc(sizeof(struct node));
tempcell->element=y;
Compiler (6) Resources Compile Log Debug Find Results Console
Line: 1 Col: 1 Sel: 0 Lines: 127 Length: 2297 Insert Done parsing in 0.015 seconds


C:\Users\durga\OneDrive\Documents\singly linked list.cpp - [Executing] - Embarcadero Dev-C++ 6.3
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 9.2.0 64-bit Release
(globals)
doubly linked list.cpp
singly linked list.cpp
return k;
}
int insert(int y,position p)
{
position tempcell;
tempcell=(position)malloc(sizeof(struct node));
tempcell->element=y;
tempcell->next=p->next;
p->next=tempcell;
}
int deletel(int y,list h)
{
position p,p1;
p=findPrevious(y,h);
if(p->next!=NULL)
{
p1=p->next;
p->next=p1->next;
free(p1);
printf("the element %d is deleted from the list",y);
}
else
printf("The element does not exist");
}
position findPrevious(int y,list h)
{
position p;
p=h;
while(p->next!=NULL && p->next->element!=y)
p=p->next;
return p;
}
int printlist(position n)
{
position l;
for(l=n->next;l!=NULL;l=l->next)
printf("-->%d-->",l->element);
}
Compiler (6) Resources Compile Log Debug Find Results Console
Line: 1 Col: 1 Sel: 0 Lines: 127 Length: 2297 Insert Done parsing in 0.015 seconds


C:\Users\durga\OneDrive\Do
do u want to perform any operation again
enter 1->yes
2->no:1
Enter the operation u want to perform with the list
1->find
2->insert
3->delete
4->display
4
HEADER-->45-->NULL
do u want to perform any operation again
enter 1->yes
2->no:1
Enter the operation u want to perform with the list
1->find
2->insert
3->delete
4->display
2
Enter the data u want to insert:2345
Enter the element after which data to be inserted45
do u want to perform any operation again
enter 1->yes
2->no:1
Enter the operation u want to perform with the list
1->find
2->insert
3->delete
4->display
4
HEADER-->45-->-->2345-->NULL
do u want to perform any operation again
enter 1->yes
2->no: