Name - Sanjoy Saha

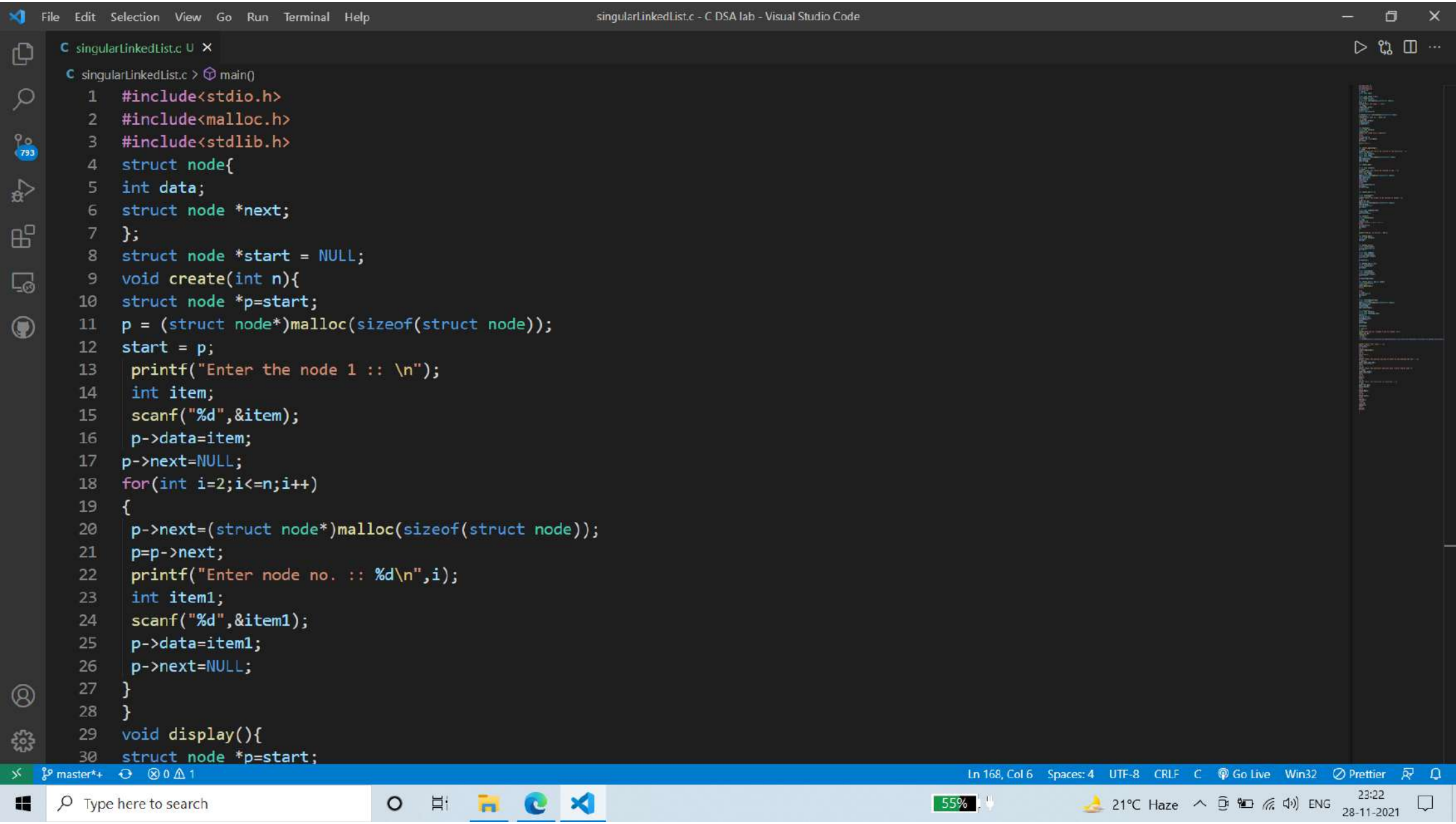
Stream - Computer Science & Engineering

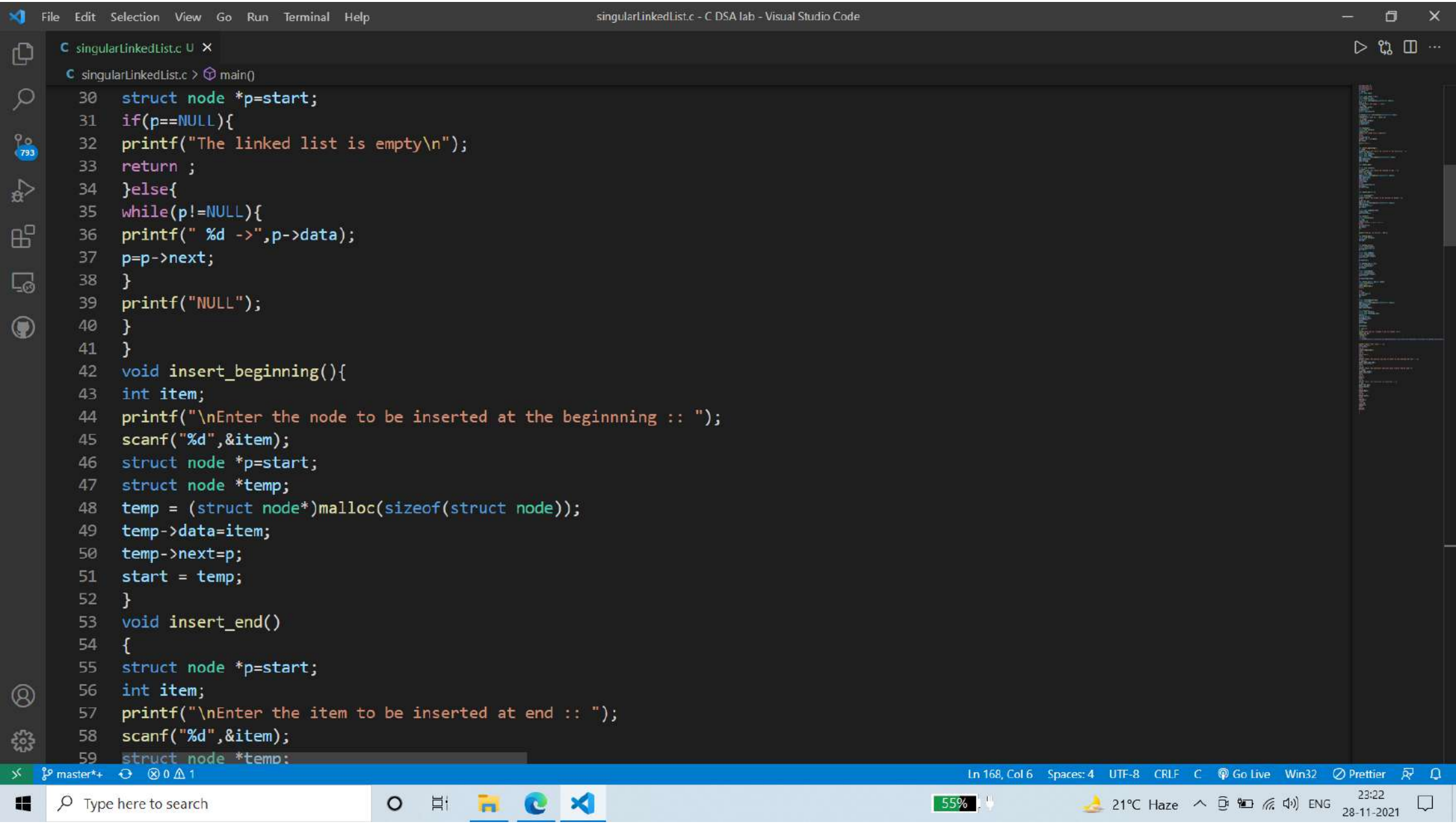
Sec - A

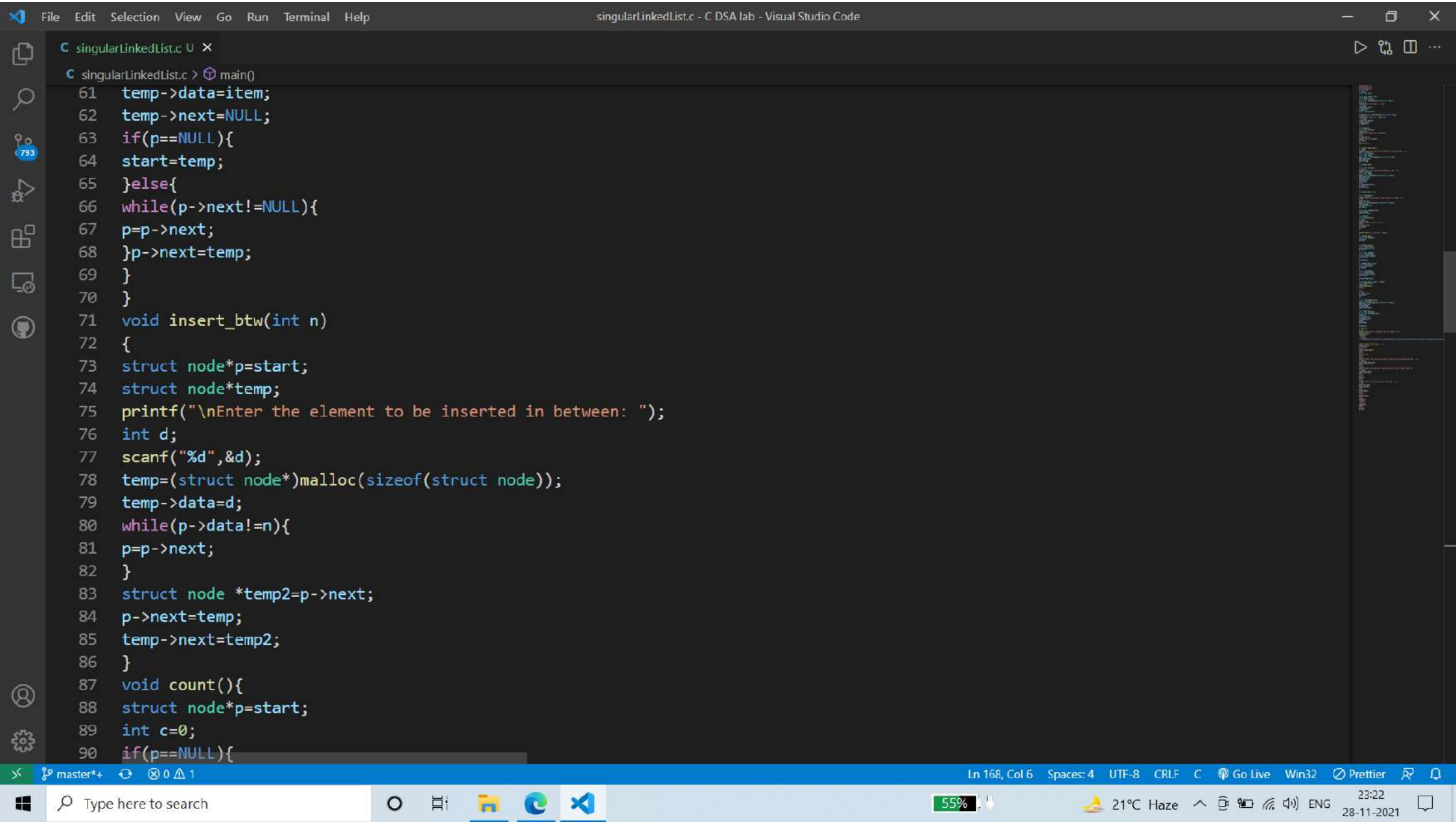
Roll no. - 3

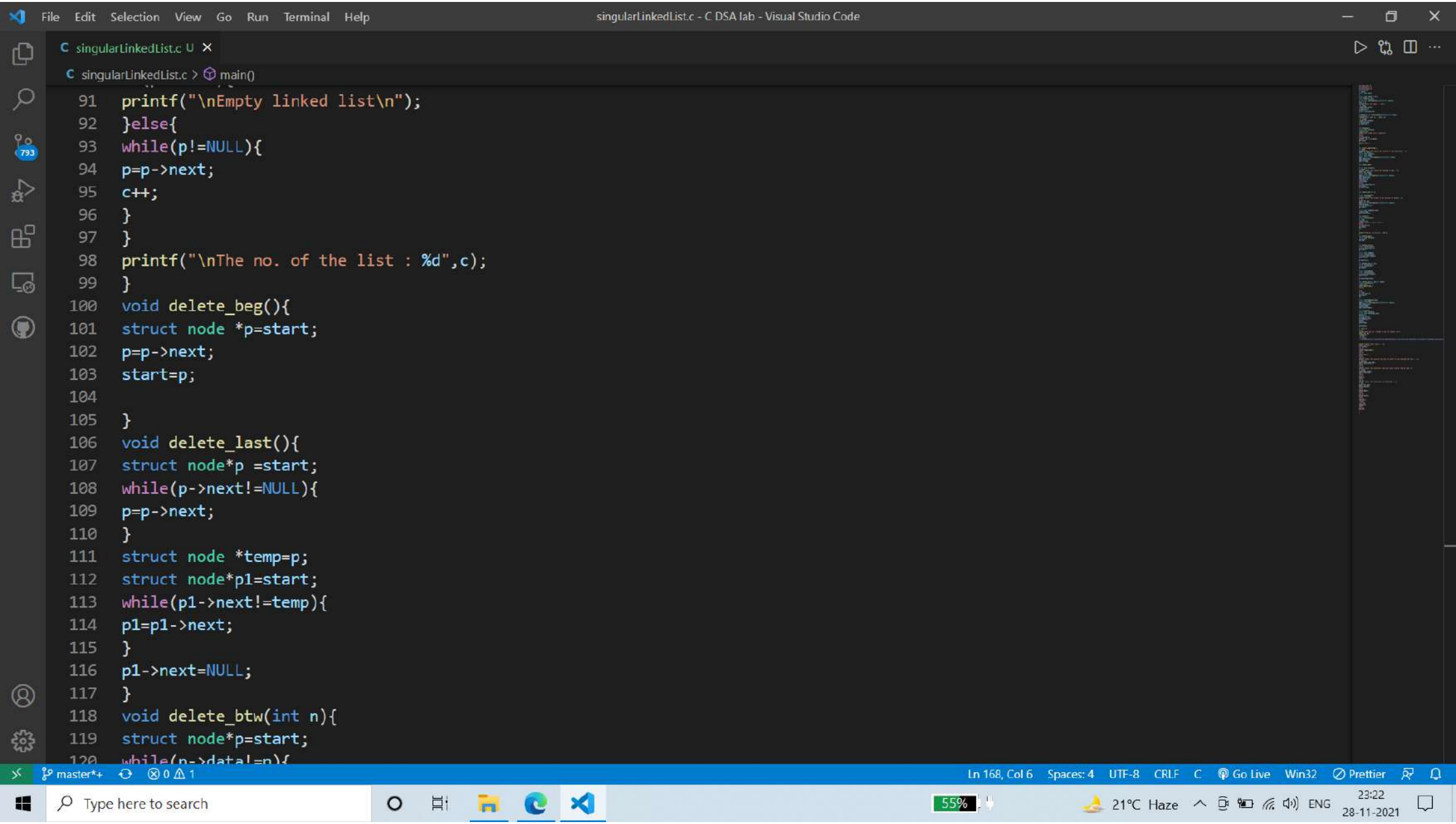
University Roll no. :- 10900120003

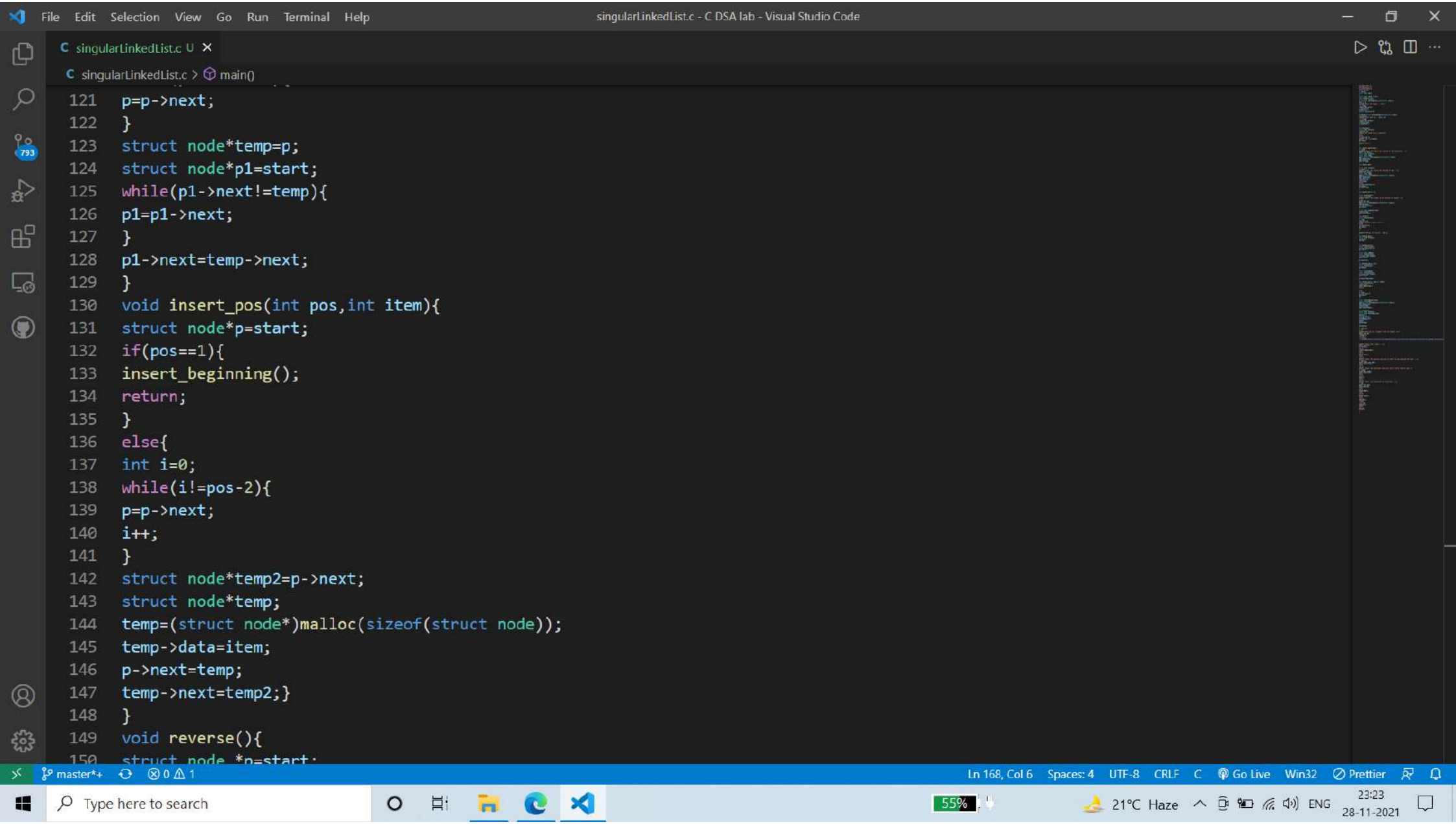
Subject - DSA Lab

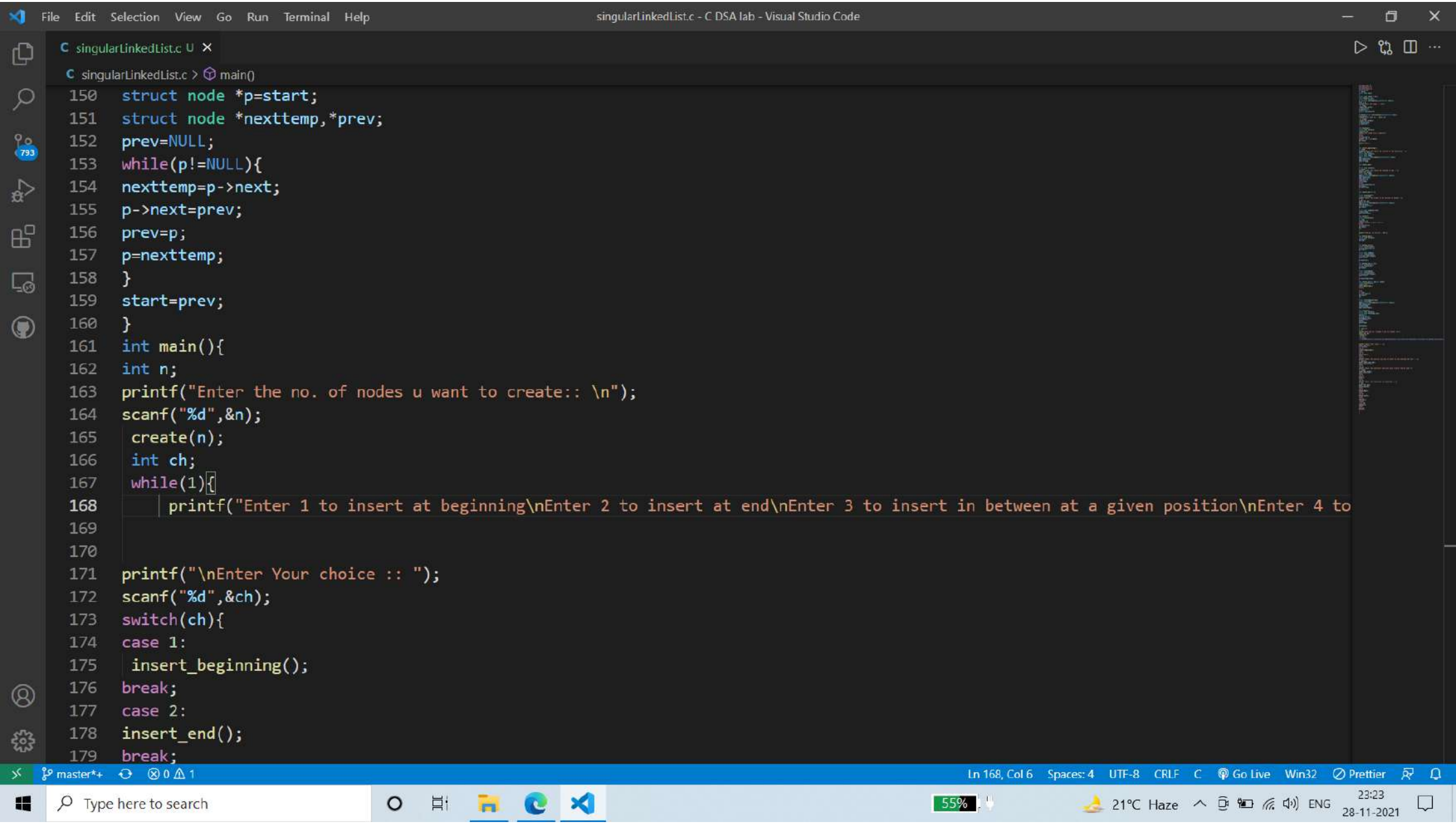

File Edit Selection View Go Run Terminal Help
singularLinkedList.c - C DSA lab - Visual Studio Code
singularLinkedList.c U
singularLinkedList.c > main()
#include<stdio.h>
#include<malloc.h>
#include<stdlib.h>
struct node{
int data;
struct node *next;
};
struct node *start = NULL;
void create(int n){
struct node *p=start;
p = (struct node*)malloc(sizeof(struct node));
start = p;
printf("Enter the node 1 :: \n");
int item;
scanf("%d",&item);
p->data=item;
p->next=NULL;
for(int i=2;i<=n;i++)
{
p->next=(struct node*)malloc(sizeof(struct node));
p=p->next;
printf("Enter node no. :: %d\n",i);
int item1;
scanf("%d",&item1);
p->data=item1;
p->next=NULL;
}
}
void display(){
struct node *p=start;
master*+ 0 1
Ln 168, Col 6 Spaces: 4 UTF-8 CRLF C Go Live Win32 Prettier
Type here to search
55%
21°C Haze
ENG
23:22
28-11-2021

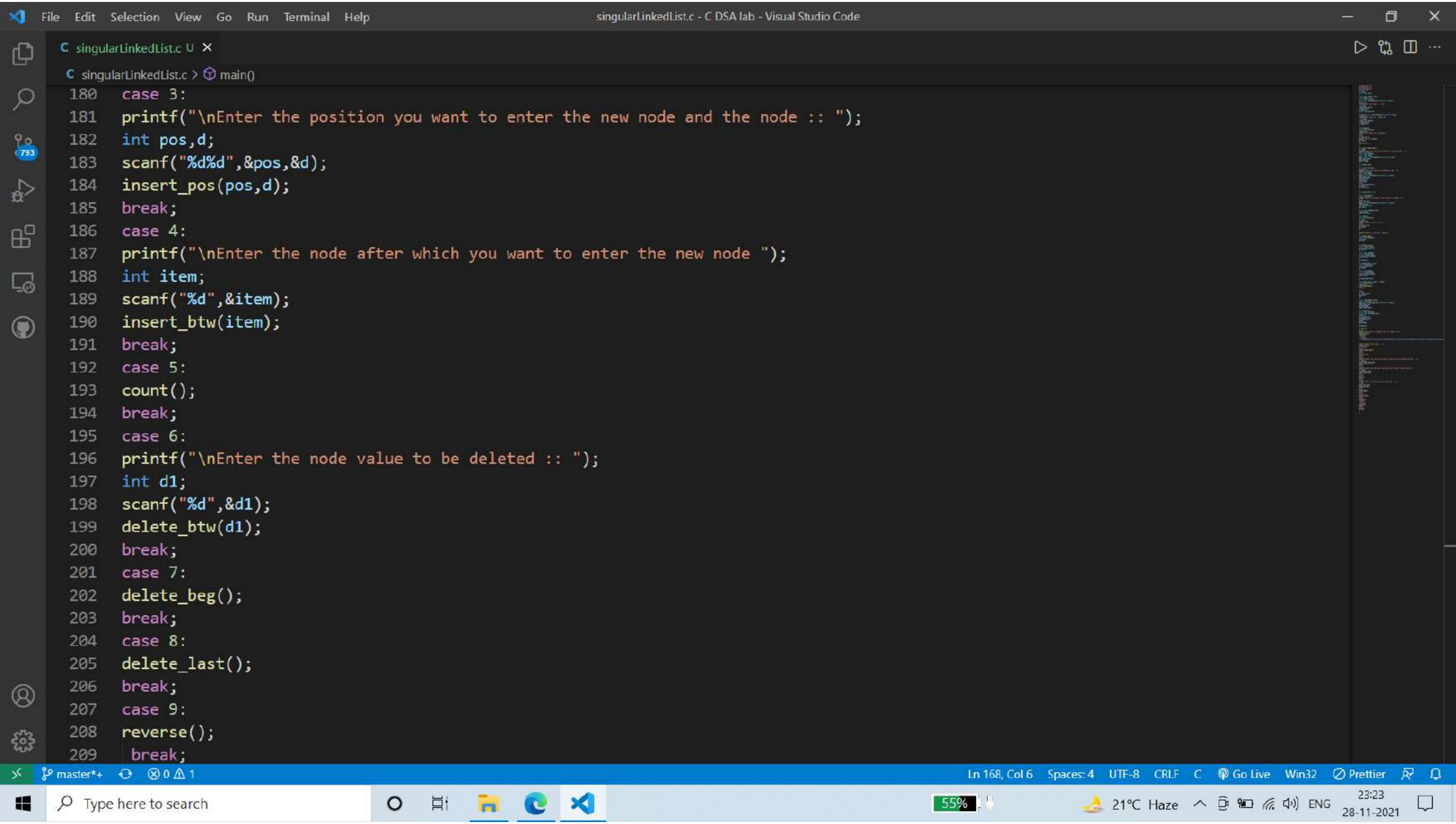

File Edit Selection View Go Run Terminal Help
singularLinkedList.c - C DSA lab - Visual Studio Code
singularLinkedList.c U
singularLinkedList.c > main()
30 struct node *p=start;
31 if(p==NULL){
32 printf("The linked list is empty\n");
33 return ;
34 }else{
35 while(p!=NULL){
36 printf(" %d ->",p->data);
37 p=p->next;
38 }
39 printf("NULL");
40 }
41 }
42 void insert_beginning(){
43 int item;
44 printf("\nEnter the node to be inserted at the beginnning :: ");
45 scanf("%d",&item);
46 struct node *p=start;
47 struct node *temp;
48 temp = (struct node*)malloc(sizeof(struct node));
49 temp->data=item;
50 temp->next=p;
51 start = temp;
52 }
53 void insert_end()
54 {
55 struct node *p=start;
56 int item;
57 printf("\nEnter the item to be inserted at end :: ");
58 scanf("%d",&item);
59 struct node *temp;
master*+ 0 1
Ln 168, Col 6 Spaces: 4 UTF-8 CRLF C Go Live Win32 Prettier
Type here to search
55%
21°C Haze
ENG
23:22
28-11-2021

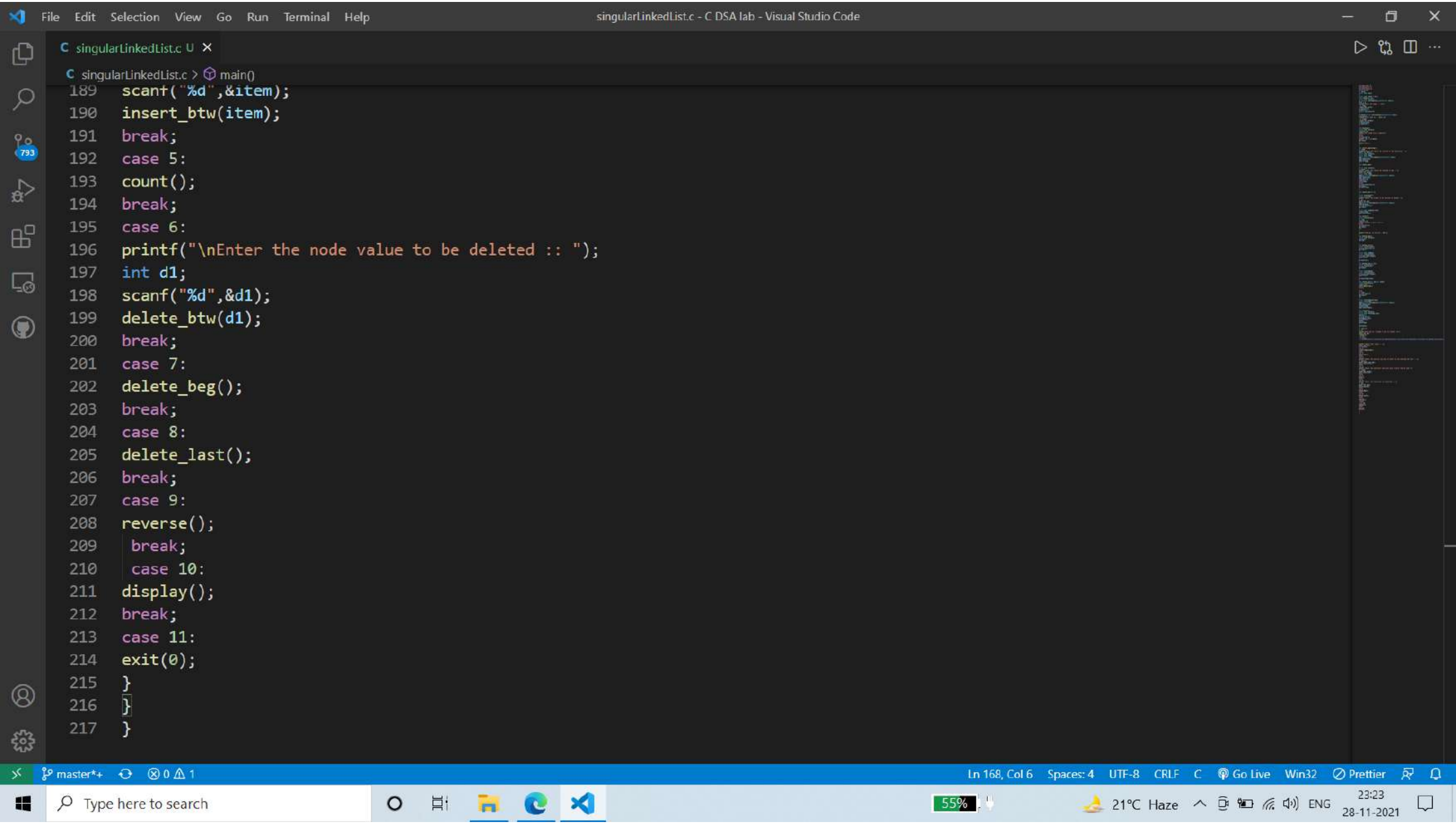

File Edit Selection View Go Run Terminal Help
singularLinkedList.c - C DSA lab - Visual Studio Code
singularLinkedList.c U
singularLinkedList.c > main()
temp->data=item;
temp->next=NULL;
if(p==NULL){
start=temp;
}else{
while(p->next!=NULL){
p=p->next;
}p->next=temp;
}
}
void insert_btw(int n)
{
struct node*p=start;
struct node*temp;
printf("\nEnter the element to be inserted in between: ");
int d;
scanf("%d",&d);
temp=(struct node*)malloc(sizeof(struct node));
temp->data=d;
while(p->data!=n){
p=p->next;
}
struct node *temp2=p->next;
p->next=temp;
temp->next=temp2;
}
void count(){
struct node*p=start;
int c=0;
if(p==NULL){
master*+
0 1
Ln 168, Col 6
Spaces: 4
UTF-8
CRLF
C
Go Live
Win32
Prettier
Type here to search
55%
21°C Haze
ENG
23:22
28-11-2021


File Edit Selection View Go Run Terminal Help
singularLinkedList.c - C DSA lab - Visual Studio Code
singularLinkedList.c U
singularLinkedList.c > main()
printf("\nEmpty linked list\n");
}else{
while(p!=NULL){
p=p->next;
c++;
}
}
printf("\nThe no. of the list : %d",c);
}
void delete_beg(){
struct node *p=start;
p=p->next;
start=p;
}
void delete_last(){
struct node*p =start;
while(p->next!=NULL){
p=p->next;
}
struct node *temp=p;
struct node*p1=start;
while(p1->next!=temp){
p1=p1->next;
}
p1->next=NULL;
}
void delete_btw(int n){
struct node*p=start;
master*+ 0 1
Ln 168, Col 6 Spaces: 4 UTF-8 CRLF C Go Live Win32 Prettier
Type here to search
55%
21°C Haze
ENG
23:22
28-11-2021


File Edit Selection View Go Run Terminal Help
singularLinkedList.c - C DSA lab - Visual Studio Code
singularLinkedList.c U
singularLinkedList.c > main()
p=p->next;
}
struct node*temp=p;
struct node*p1=start;
while(p1->next!=temp){
p1=p1->next;
}
p1->next=temp->next;
}
void insert_pos(int pos,int item){
struct node*p=start;
if(pos==1){
insert_beginning();
return;
}
else{
int i=0;
while(i!=pos-2){
p=p->next;
i++;
}
struct node*temp2=p->next;
struct node*temp;
temp=(struct node*)malloc(sizeof(struct node));
temp->data=item;
p->next=temp;
temp->next=temp2;}
}
void reverse(){
struct node *p=start;
master*+ 0 1
Ln 168, Col 6 Spaces: 4 UTF-8 CRLF C Go Live Win32 Prettier
Type here to search
55%
21°C Haze
ENG
23:23
28-11-2021


File Edit Selection View Go Run Terminal Help
singularLinkedList.c - C DSA lab - Visual Studio Code
singularLinkedList.c U
singularLinkedList.c > main()
150 struct node *p=start;
151 struct node *nexttemp,*prev;
152 prev=NULL;
153 while(p!=NULL){
154 nexttemp=p->next;
155 p->next=prev;
156 prev=p;
157 p=nexttemp;
158 }
159 start=prev;
160 }
161 int main(){
162 int n;
163 printf("Enter the no. of nodes u want to create:: \n");
164 scanf("%d",&n);
165 create(n);
166 int ch;
167 while(1){
168 printf("Enter 1 to insert at beginning\nEnter 2 to insert at end\nEnter 3 to insert in between at a given position\nEnter 4 to
169
170
171 printf("\nEnter Your choice :: ");
172 scanf("%d",&ch);
173 switch(ch){
174 case 1:
175 insert_beginning();
176 break;
177 case 2:
178 insert_end();
179 break;
master*+ 0 1
Ln 168, Col 6 Spaces: 4 UTF-8 CRLF C Go Live Win32 Prettier
Type here to search
55%
21°C Haze
ENG
23:23
28-11-2021


File Edit Selection View Go Run Terminal Help
singularLinkedList.c - C DSA lab - Visual Studio Code
singularLinkedList.c U
singularLinkedList.c > main()
case 3:
printf("\nEnter the position you want to enter the new node and the node :: ");
int pos,d;
scanf("%d%d",&pos,&d);
insert_pos(pos,d);
break;
case 4:
printf("\nEnter the node after which you want to enter the new node ");
int item;
scanf("%d",&item);
insert_btw(item);
break;
case 5:
count();
break;
case 6:
printf("\nEnter the node value to be deleted :: ");
int d1;
scanf("%d",&d1);
delete_btw(d1);
break;
case 7:
delete_beg();
break;
case 8:
delete_last();
break;
case 9:
reverse();
break;
master*+ 0 1
Ln 168, Col 6 Spaces: 4 UTF-8 CRLF C Go Live Win32 Prettier
Type here to search
55%
21°C Haze
ENG
23:23
28-11-2021


File Edit Selection View Go Run Terminal Help
singularLinkedList.c - C DSA lab - Visual Studio Code
singularLinkedList.c U
singularLinkedList.c > main()
scanf("%d",&item);
insert_btw(item);
break;
case 5:
count();
break;
case 6:
printf("\nEnter the node value to be deleted :: ");
int d1;
scanf("%d",&d1);
delete_btw(d1);
break;
case 7:
delete_beg();
break;
case 8:
delete_last();
break;
case 9:
reverse();
break;
case 10:
display();
break;
case 11:
exit(0);
}
}
}
master*+ 0 1
Ln 168, Col 6 Spaces: 4 UTF-8 CRLF C Go Live Win32 Prettier
Type here to search
55%
21°C Haze
ENG
23:23
28-11-2021

PROBLEMS 1 OUTPUT TERMINAL DEBUG CONSOLE

```
Enter 10 to display the list
Enter 11 to exit


Enter Your choice :: 10
 434 -> 121 -> 33 -> 3 -> 4 -> 2 ->NULLEnter 1 to insert at beginning
Enter 2 to insert at end
Enter 3 to insert in between at a given position
Enter 4 to insert in between after a given node
Enter 5 to count the number of the list
Enter 6 to delete a node with a given value
Enter 7 to delete first node
Enter 8 to delete last node
Enter 9 to reverse the list
Enter 10 to display the list
Enter 11 to exit


Enter Your choice :: 9
Enter 1 to insert at beginning
Enter 2 to insert at end
Enter 3 to insert in between at a given position
Enter 4 to insert in between after a given node
Enter 5 to count the number of the list
Enter 6 to delete a node with a given value
Enter 7 to delete first node
Enter 8 to delete last node
Enter 9 to reverse the list
Enter 10 to display the list
Enter 11 to exit


Enter Your choice :: 10
 2 -> 4 -> 3 -> 33 -> 121 -> 434 ->NULLEnter 1 to insert at beginning
Enter 2 to insert at end
Enter 3 to insert in between at a given position
Enter 4 to insert in between after a given node
Enter 5 to count the number of the list
Enter 6 to delete a node with a given value
Enter 7 to delete first node
Enter 8 to delete last node
Enter 9 to reverse the list
Enter 10 to display the list
Enter 11 to exit
```

PROBLEMS 1 OUTPUT TERMINAL DEBUG CONSOLE

```
Enter Your choice :: 8
Enter 1 to insert at beginning
Enter 2 to insert at end
Enter 3 to insert in between at a given position
Enter 4 to insert in between after a given node
Enter 5 to count the number of the list
Enter 6 to delete a node with a given value
Enter 7 to delete first node
Enter 8 to delete last node
Enter 9 to reverse the list
Enter 10 to display the list
Enter 11 to exit


Enter Your choice :: 10
 2 -> 4 -> 3 -> 33 -> 121 ->NULLEnter 1 to insert at beginning
Enter 2 to insert at end
Enter 3 to insert in between at a given position
Enter 4 to insert in between after a given node
Enter 5 to count the number of the list
Enter 6 to delete a node with a given value
Enter 7 to delete first node
Enter 8 to delete last node
Enter 9 to reverse the list
Enter 10 to display the list
Enter 11 to exit


Enter Your choice :: 7
Enter 1 to insert at beginning
Enter 2 to insert at end
Enter 3 to insert in between at a given position
Enter 4 to insert in between after a given node
Enter 5 to count the number of the list
Enter 6 to delete a node with a given value
Enter 7 to delete first node
Enter 8 to delete last node
Enter 9 to reverse the list
Enter 10 to display the list
Enter 11 to exit
```

PROBLEMS 1 OUTPUT TERMINAL DEBUG CONSOLE

```
Enter 11 to exit


Enter Your choice :: 7
Enter 1 to insert at beginning
Enter 2 to insert at end
Enter 3 to insert in between at a given position
Enter 4 to insert in between after a given node
Enter 5 to count the number of the list
Enter 6 to delete a node with a given value
Enter 7 to delete first node
Enter 8 to delete last node
Enter 9 to reverse the list
Enter 10 to display the list
Enter 11 to exit


Enter Your choice :: 10
 4 -> 3 -> 33 -> 121 ->NULLEnter 1 to insert at beginning
Enter 2 to insert at end
Enter 3 to insert in between at a given position
Enter 4 to insert in between after a given node
Enter 5 to count the number of the list
Enter 6 to delete a node with a given value
Enter 7 to delete first node
Enter 8 to delete last node
Enter 9 to reverse the list
Enter 10 to display the list
Enter 11 to exit


Enter Your choice :: 5

The no. of the list : 4Enter 1 to insert at beginning
Enter 2 to insert at end
Enter 3 to insert in between at a given position
Enter 4 to insert in between after a given node
Enter 5 to count the number of the list
Enter 6 to delete a node with a given value
Enter 7 to delete first node
Enter 8 to delete last node
Enter 9 to reverse the list
Enter 10 to display the list
Enter 11 to exit
```

PROBLEMS 1 OUTPUT TERMINAL DEBUG CONSOLE

```
Enter 10 to display the list
Enter 11 to exit


Enter Your choice :: 10
 4 -> 3 -> 33 -> 121 ->NULLEnter 1 to insert at beginning
Enter 2 to insert at end
Enter 3 to insert in between at a given position
Enter 4 to insert in between after a given node
Enter 5 to count the number of the list
Enter 6 to delete a node with a given value
Enter 7 to delete first node
Enter 8 to delete last node
Enter 9 to reverse the list
Enter 10 to display the list
Enter 11 to exit


Enter Your choice :: 5

The no. of the list : 4Enter 1 to insert at beginning
Enter 2 to insert at end
Enter 3 to insert in between at a given position
Enter 4 to insert in between after a given node
Enter 5 to count the number of the list
Enter 6 to delete a node with a given value
Enter 7 to delete first node
Enter 8 to delete last node
Enter 9 to reverse the list
Enter 10 to display the list
Enter 11 to exit


Enter Your choice :: 10
 4 -> 3 -> 33 -> 121 ->NULLEnter 1 to insert at beginning
Enter 2 to insert at end
Enter 3 to insert in between at a given position
Enter 4 to insert in between after a given node
Enter 5 to count the number of the list
Enter 6 to delete a node with a given value
Enter 7 to delete first node
Enter 8 to delete last node
Enter 9 to reverse the list
Enter 10 to display the list
Enter 11 to exit
```