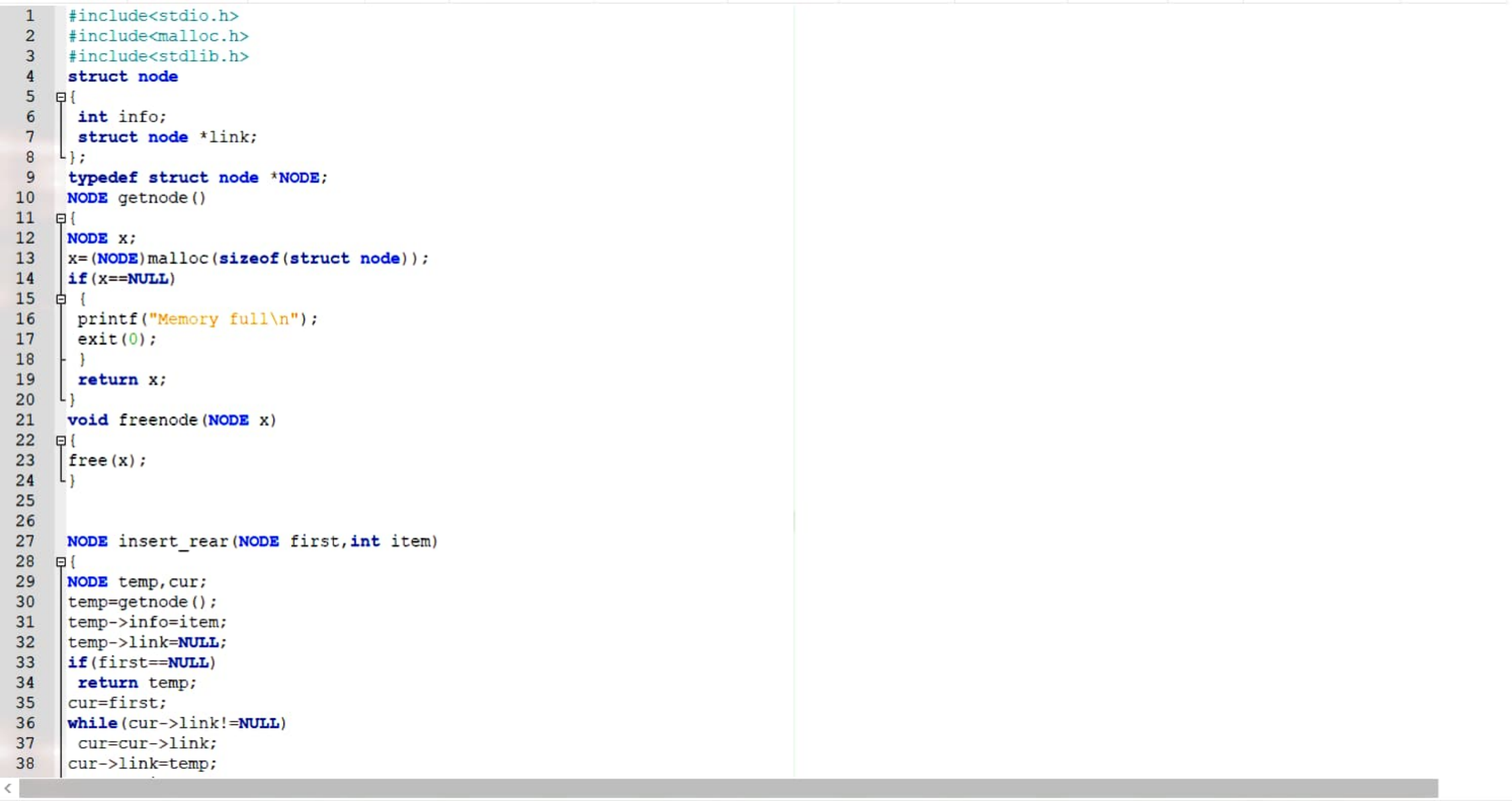
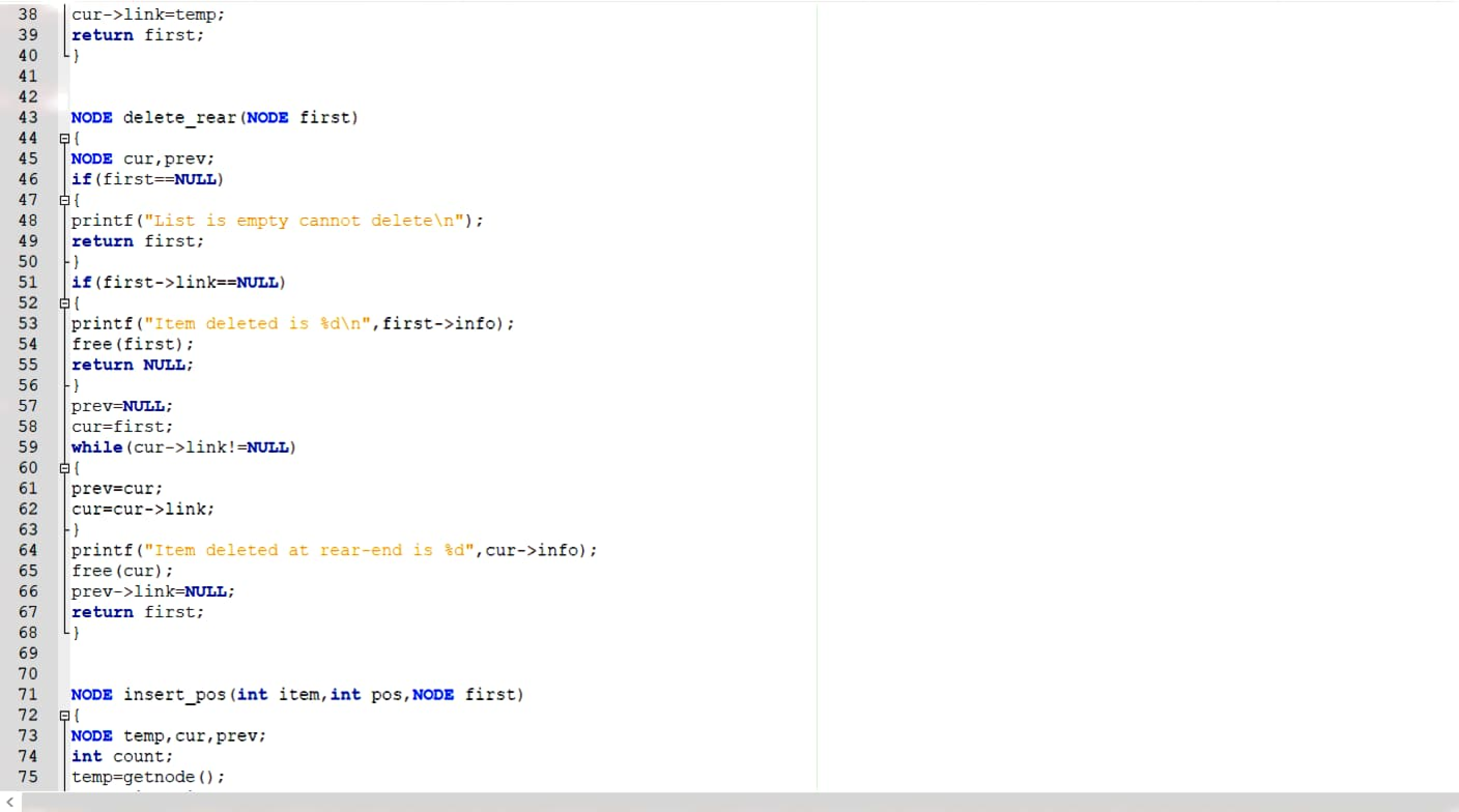
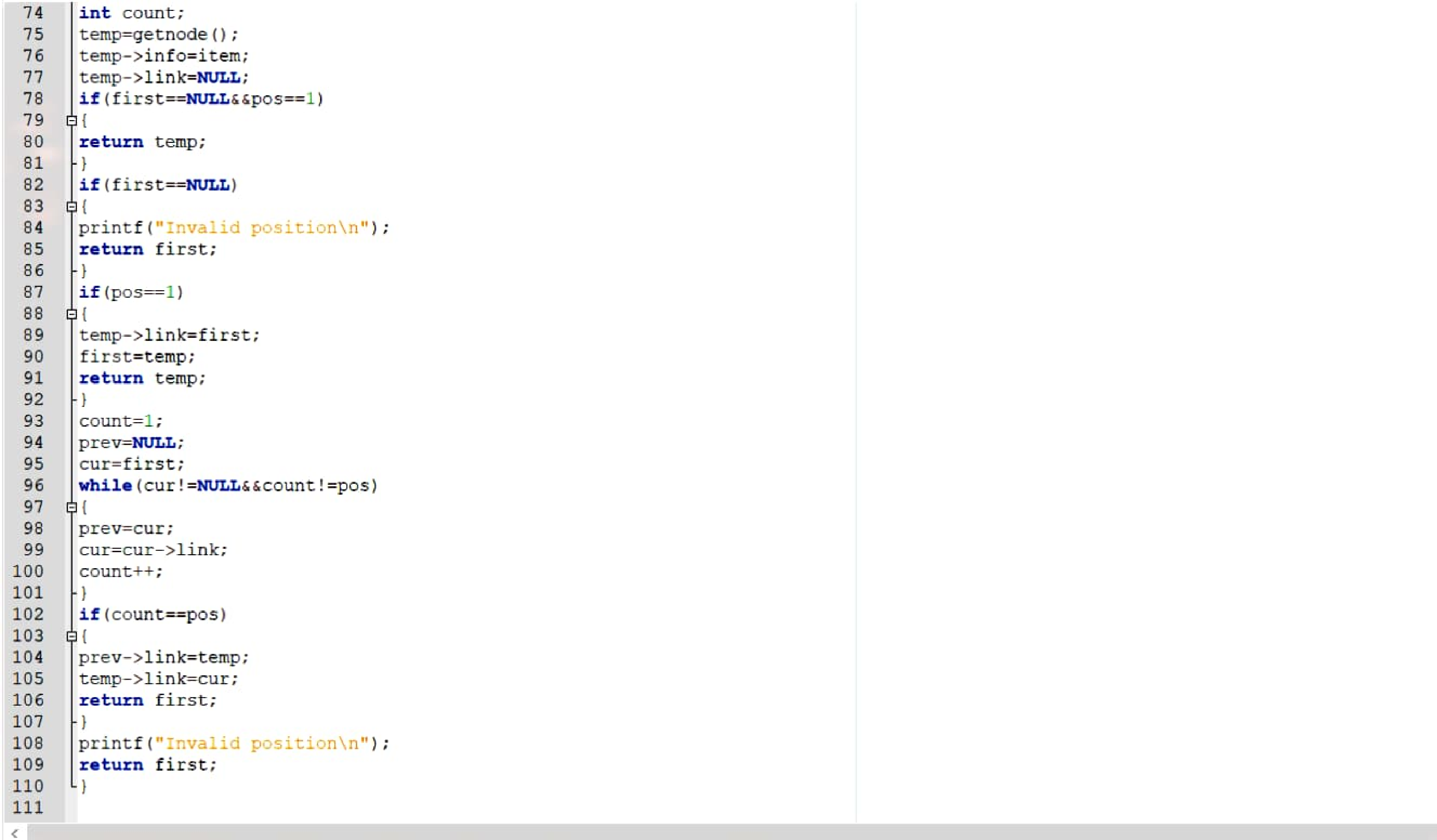


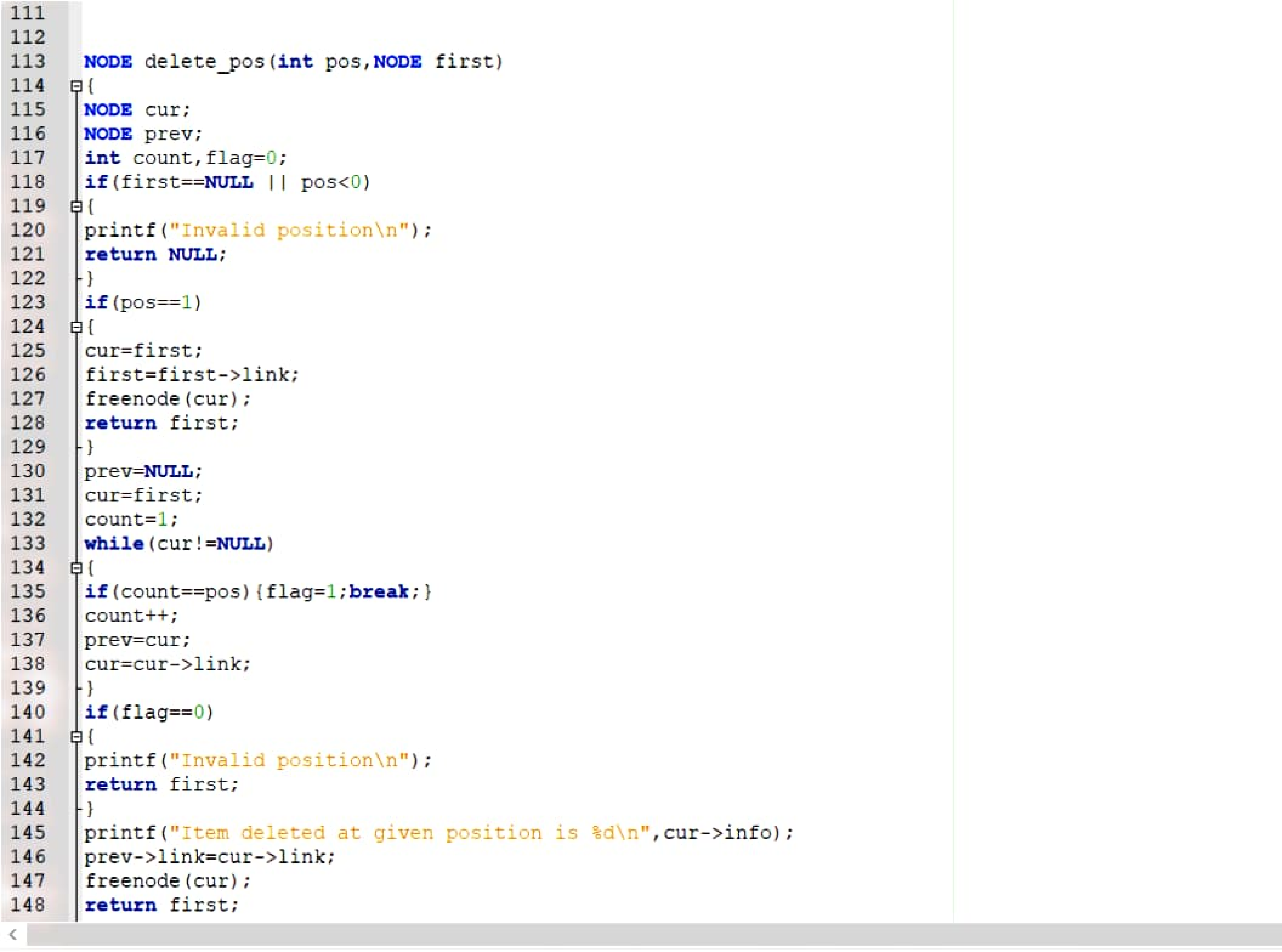
```c
#include<stdio.h>
#include<malloc.h>
#include<stdlib.h>
struct node
{
 int info;
 struct node *link;
};
typedef struct node *NODE;
NODE getnode()
{
NODE x;
x=(NODE)malloc(sizeof(struct node));
if(x==NULL)
 {
 printf("Memory full\n");
 exit(0);
 }
 return x;
}
void freenode(NODE x)
{
free(x);
}


NODE insert_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;
```



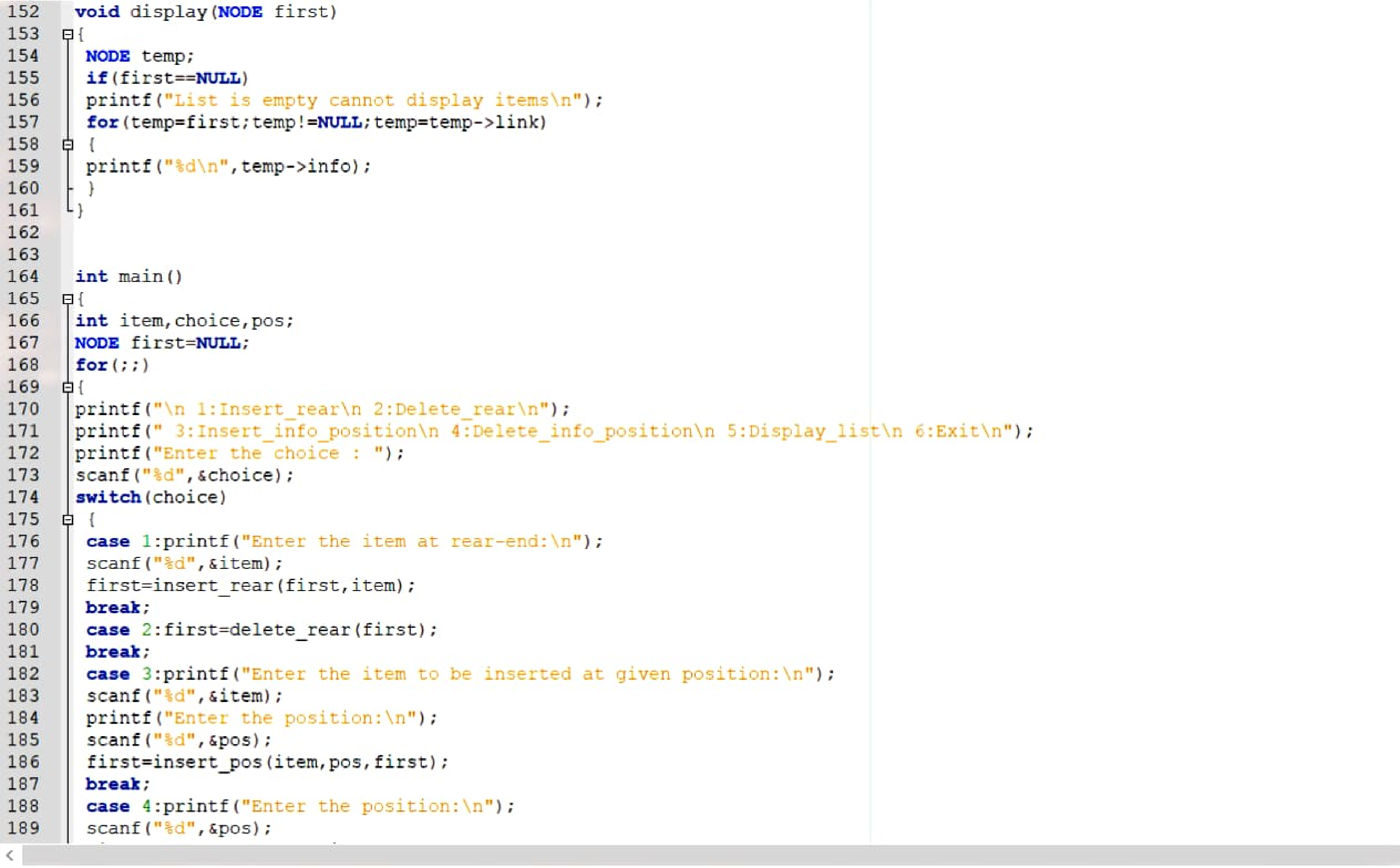
```c
cur->link=temp;
return first;
}


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 insert_pos(int item,int pos,NODE first)
{
NODE temp,cur,prev;
int count;
temp=getnode();
```



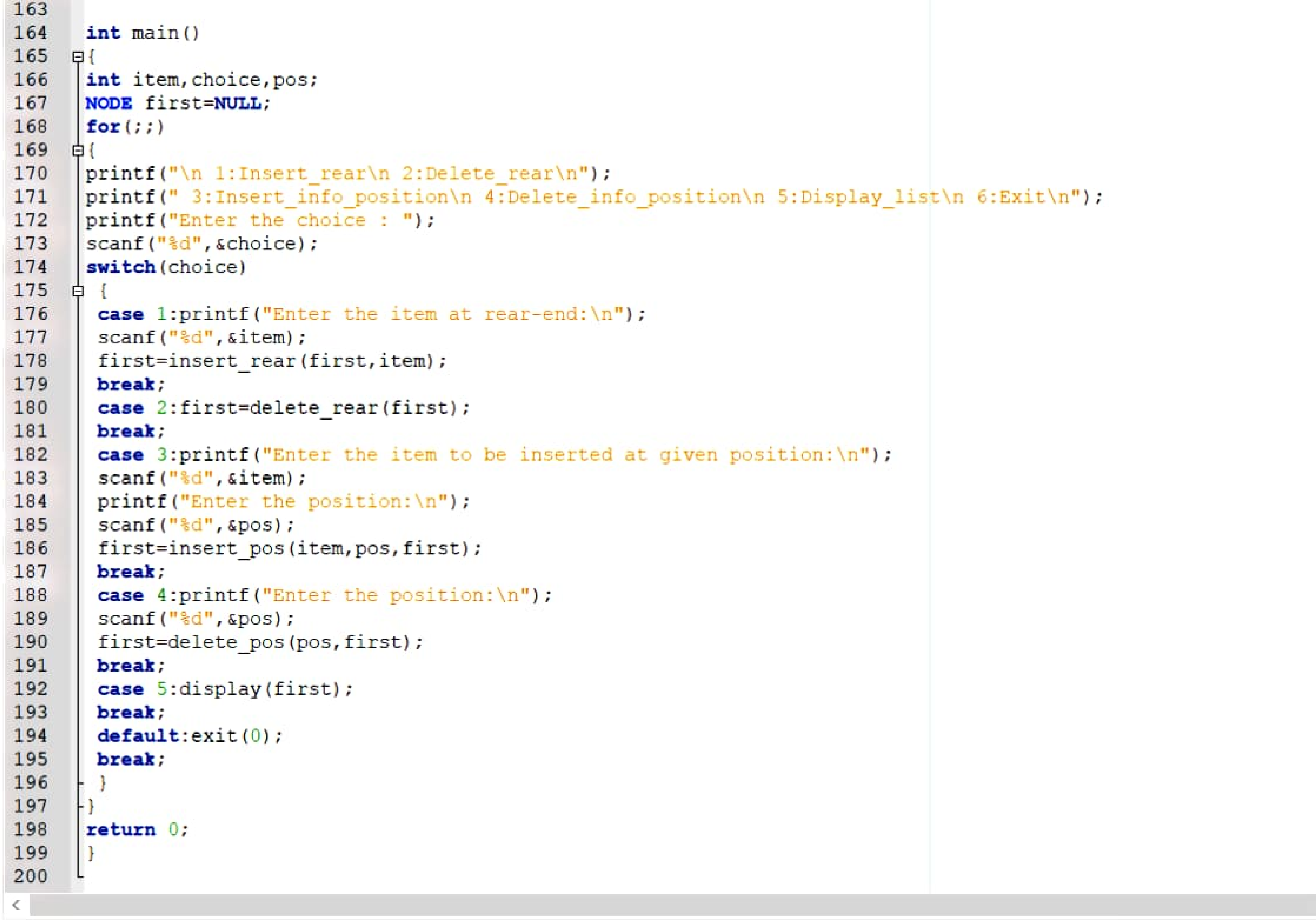
```c
int count;
temp=getnode();
temp->info=item;
temp->link=NULL;
if(first==NULL&&pos==1)
{
return temp;
}
if(first==NULL)
{
printf("Invalid position\n");
return first;
}
if(pos==1)
{
temp->link=first;
first=temp;
return temp;
}
count=1;
prev=NULL;
cur=first;
while(cur!=NULL&&count!=pos)
{
prev=cur;
cur=cur->link;
count++;
}
if(count==pos)
{
prev->link=temp;
temp->link=cur;
return first;
}
printf("Invalid position\n");
return first;
}
```



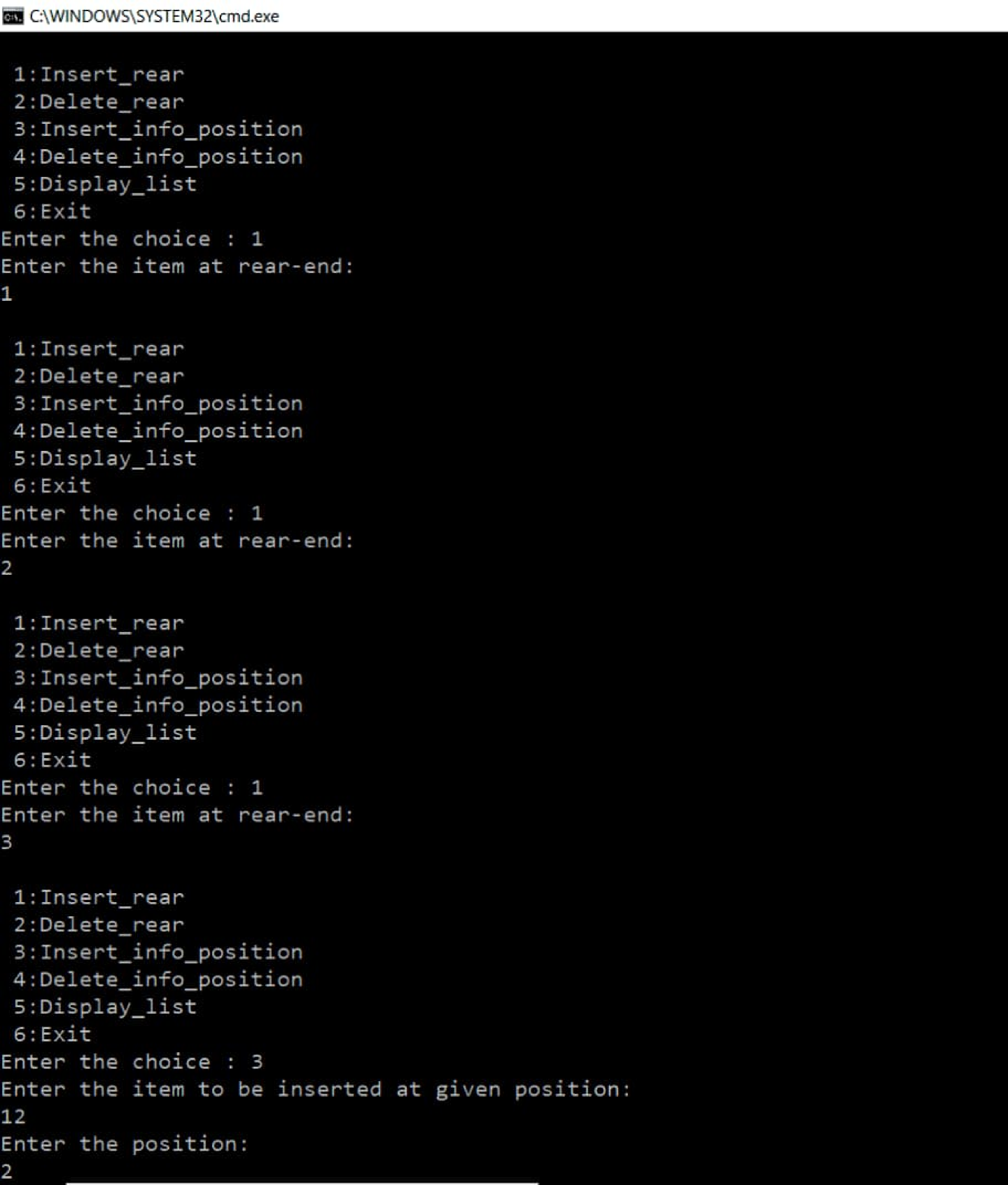
```
NODE delete_pos(int pos,NODE first)
{
NODE cur;
NODE prev;
int count,flag=0;
if(first==NULL || pos<0)
{
printf("Invalid position\n");
return NULL;
}
if(pos==1)
{
cur=first;
first=first->link;
freenode(cur);
return first;
}
prev=NULL;
cur=first;
count=1;
while(cur!=NULL)
{
if(count==pos){flag=1;break;}
count++;
prev=cur;
cur=cur->link;
}
if(flag==0)
{
printf("Invalid position\n");
return first;
}
printf("Item deleted at given position is %d\n",cur->info);
prev->link=cur->link;
freenode(cur);
return first;
```



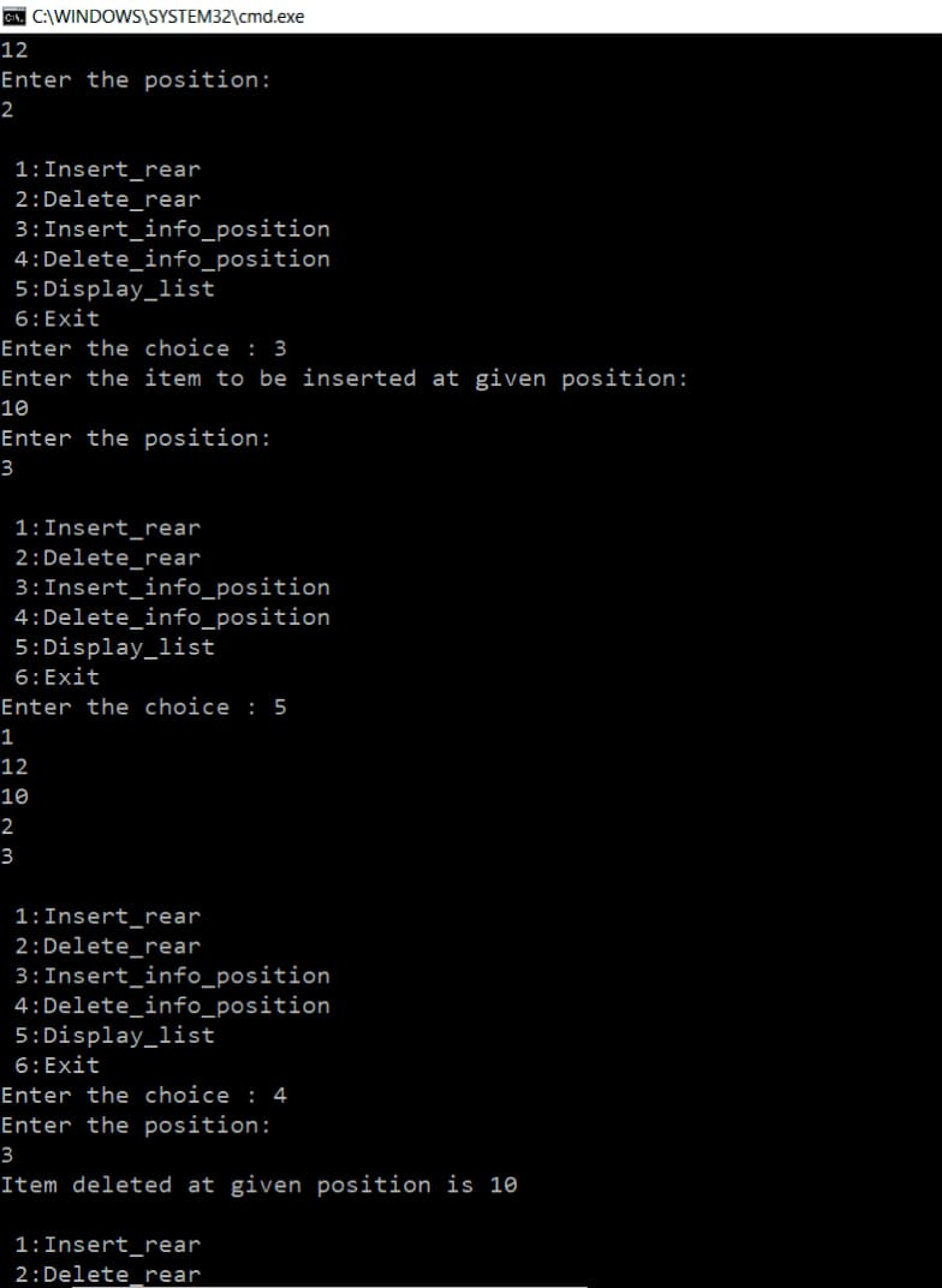
```c
void display(NODE first)
{
 NODE temp;
 if(first==NULL)
 printf("List is empty cannot display items\n");
 for(temp=first;temp!=NULL;temp=temp->link)
 {
 printf("%d\n",temp->info);
 }
}


int main()
{
int item,choice,pos;
NODE first=NULL;
for(;;)
{
printf("\n 1:Insert_rear\n 2:Delete_rear\n");
printf(" 3:Insert_info_position\n 4:Delete_info_position\n 5:Display_list\n 6:Exit\n");
printf("Enter the choice : ");
scanf("%d",&choice);
switch(choice)
 {
 case 1:printf("Enter the item at rear-end:\n");
 scanf("%d",&item);
 first=insert_rear(first,item);
 break;
 case 2:first=delete_rear(first);
 break;
 case 3:printf("Enter the item to be inserted at given position:\n");
 scanf("%d",&item);
 printf("Enter the position:\n");
 scanf("%d",&pos);
 first=insert_pos(item,pos,first);
 break;
 case 4:printf("Enter the position:\n");
 scanf("%d",&pos);
```



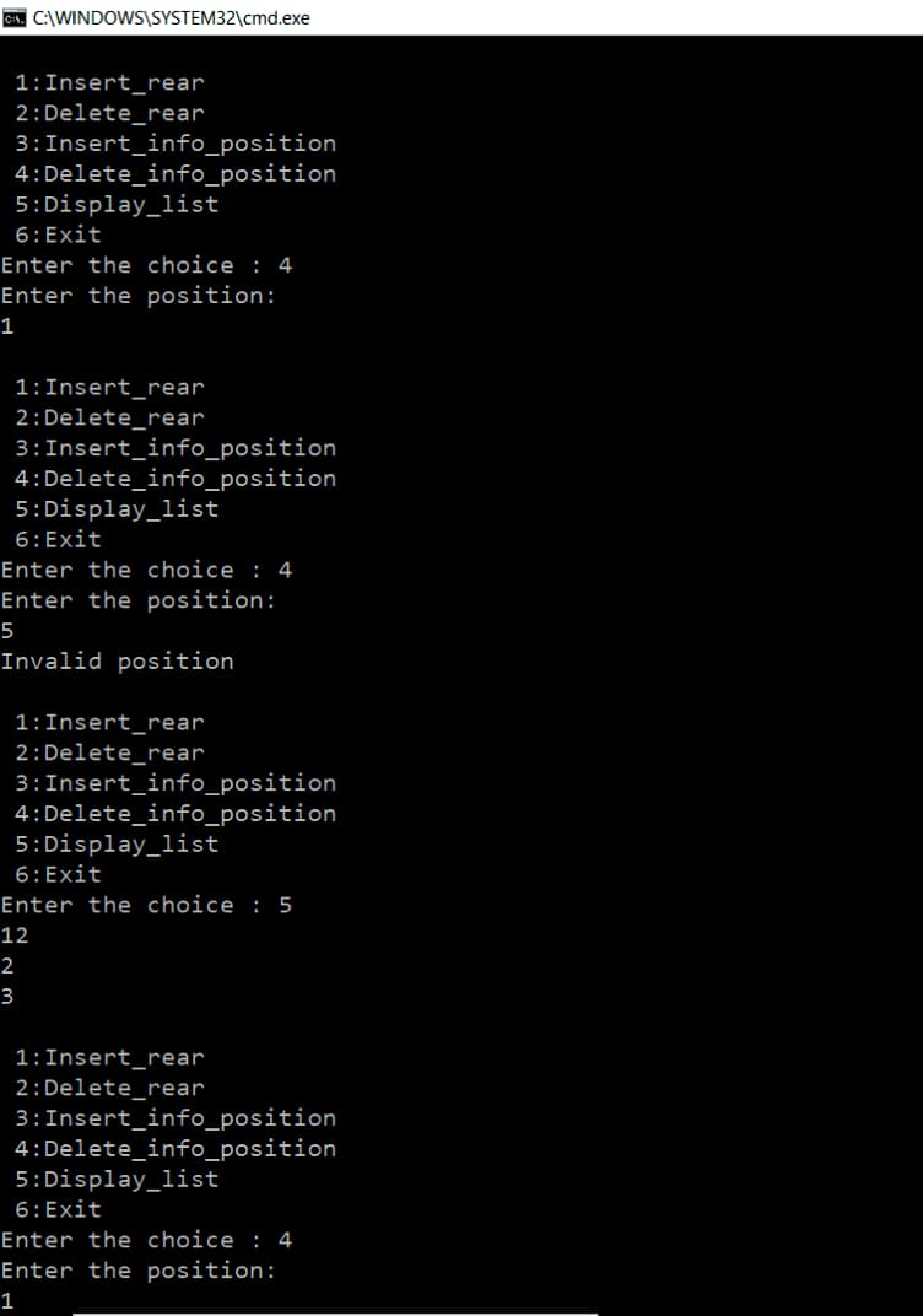
```c
int main()
{
int item,choice,pos;
NODE first=NULL;
for(;;)
{
printf("\n 1:Insert_rear\n 2:Delete_rear\n");
printf(" 3:Insert_info_position\n 4:Delete_info_position\n 5:Display_list\n 6:Exit\n");
printf("Enter the choice : ");
scanf("%d",&choice);
switch(choice)
 {
 case 1:printf("Enter the item at rear-end:\n");
 scanf("%d",&item);
 first=insert_rear(first,item);
 break;
 case 2:first=delete_rear(first);
 break;
 case 3:printf("Enter the item to be inserted at given position:\n");
 scanf("%d",&item);
 printf("Enter the position:\n");
 scanf("%d",&pos);
 first=insert_pos(item,pos,first);
 break;
 case 4:printf("Enter the position:\n");
 scanf("%d",&pos);
 first=delete_pos(pos,first);
 break;
 case 5:display(first);
 break;
 default:exit(0);
 break;
 }
}
return 0;
}
```



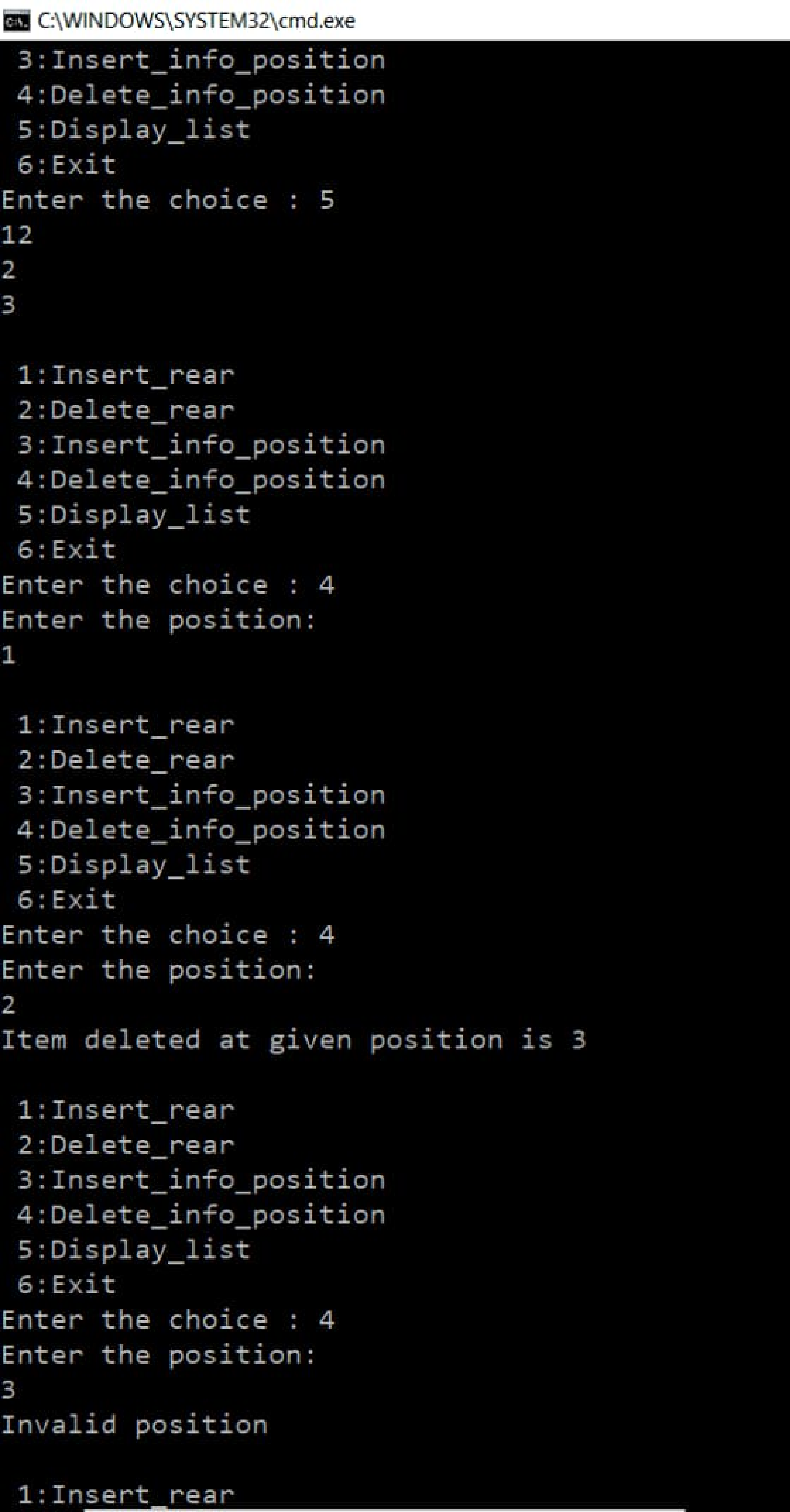
```
C:\WINDOWS\SYSTEM32\cmd.exe

 1:Insert_rear
 2:Delete_rear
 3:Insert_info_position
 4:Delete_info_position
 5:Display_list
 6:Exit
Enter the choice : 1
Enter the item at rear-end:
1

 1:Insert_rear
 2:Delete_rear
 3:Insert_info_position
 4:Delete_info_position
 5:Display_list
 6:Exit
Enter the choice : 1
Enter the item at rear-end:
2

 1:Insert_rear
 2:Delete_rear
 3:Insert_info_position
 4:Delete_info_position
 5:Display_list
 6:Exit
Enter the choice : 1
Enter the item at rear-end:
3

 1:Insert_rear
 2:Delete_rear
 3:Insert_info_position
 4:Delete_info_position
 5:Display_list
 6:Exit
Enter the choice : 3
Enter the item to be inserted at given position:
12
Enter the position:
2
```


C:\WINDOWS\SYSTEM32\cmd.exe

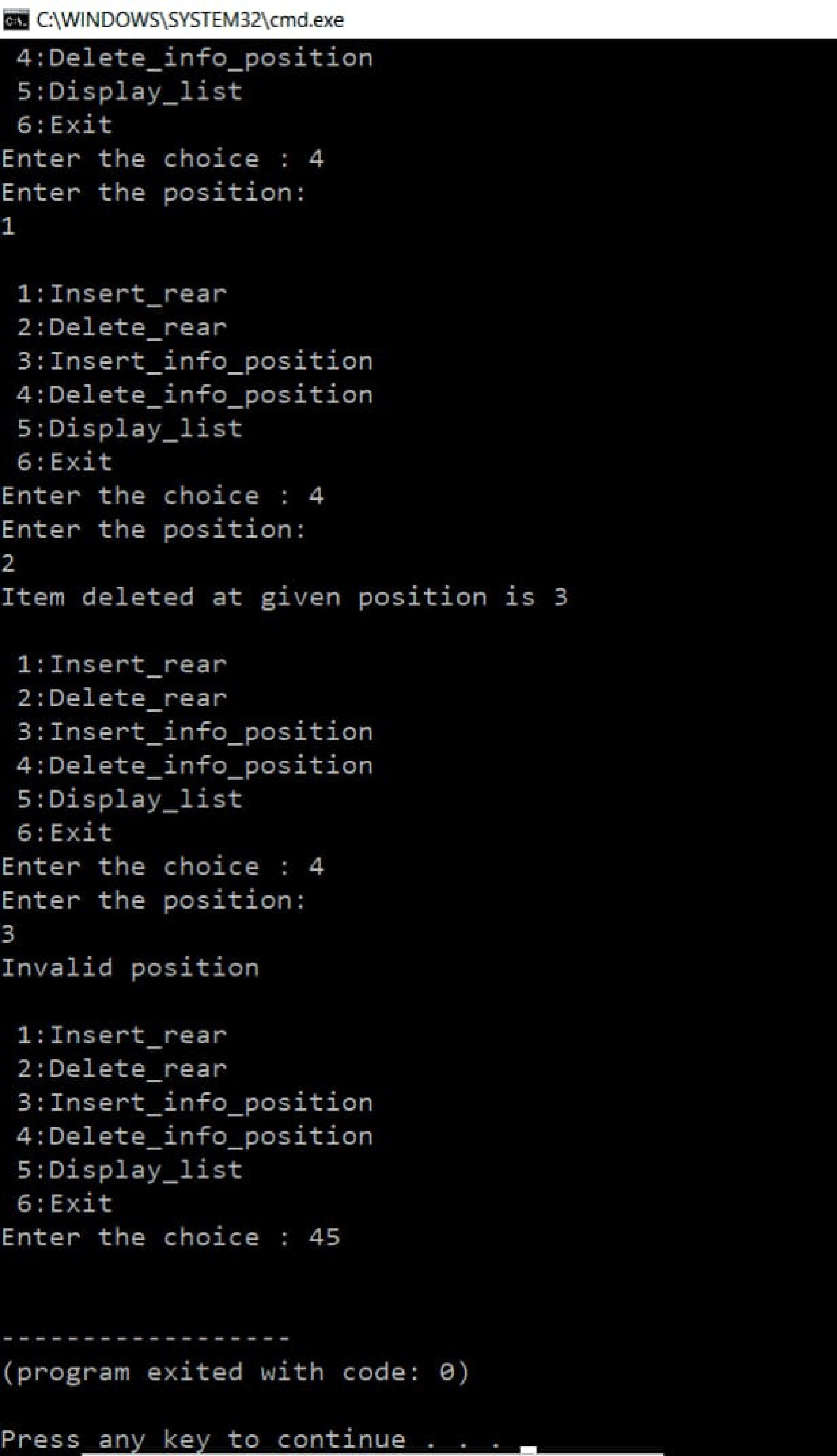
```
12
Enter the position:
2

 1:Insert_rear
 2:Delete_rear
 3:Insert_info_position
 4:Delete_info_position
 5:Display_list
 6:Exit
Enter the choice : 3
Enter the item to be inserted at given position:
10
Enter the position:
3

 1:Insert_rear
 2:Delete_rear
 3:Insert_info_position
 4:Delete_info_position
 5:Display_list
 6:Exit
Enter the choice : 5
1
12
10
2
3

 1:Insert_rear
 2:Delete_rear
 3:Insert_info_position
 4:Delete_info_position
 5:Display_list
 6:Exit
Enter the choice : 4
Enter the position:
3
Item deleted at given position is 10

 1:Insert_rear
 2:Delete_rear
```



```
C:\WINDOWS\SYSTEM32\cmd.exe

 1:Insert_rear
 2:Delete_rear
 3:Insert_info_position
 4:Delete_info_position
 5:Display_list
 6:Exit
Enter the choice : 4
Enter the position:
1

 1:Insert_rear
 2:Delete_rear
 3:Insert_info_position
 4:Delete_info_position
 5:Display_list
 6:Exit
Enter the choice : 4
Enter the position:
5
Invalid position

 1:Insert_rear
 2:Delete_rear
 3:Insert_info_position
 4:Delete_info_position
 5:Display_list
 6:Exit
Enter the choice : 5
12
2
3

 1:Insert_rear
 2:Delete_rear
 3:Insert_info_position
 4:Delete_info_position
 5:Display_list
 6:Exit
Enter the choice : 4
Enter the position:
1
```



```
C:\WINDOWS\SYSTEM32\cmd.exe
 3:Insert_info_position
 4:Delete_info_position
 5:Display_list
 6:Exit
Enter the choice : 5
12
2
3

 1:Insert_rear
 2:Delete_rear
 3:Insert_info_position
 4:Delete_info_position
 5:Display_list
 6:Exit
Enter the choice : 4
Enter the position:
1

 1:Insert_rear
 2:Delete_rear
 3:Insert_info_position
 4:Delete_info_position
 5:Display_list
 6:Exit
Enter the choice : 4
Enter the position:
2
Item deleted at given position is 3

 1:Insert_rear
 2:Delete_rear
 3:Insert_info_position
 4:Delete_info_position
 5:Display_list
 6:Exit
Enter the choice : 4
Enter the position:
3
Invalid position

 1:Insert_rear
```



C:\WINDOWS\SYSTEM32\cmd.exe

```
 4:Delete_info_position
 5:Display_list
 6:Exit
Enter the choice : 4
Enter the position:
1

 1:Insert_rear
 2:Delete_rear
 3:Insert_info_position
 4:Delete_info_position
 5:Display_list
 6:Exit
Enter the choice : 4
Enter the position:
2
Item deleted at given position is 3

 1:Insert_rear
 2:Delete_rear
 3:Insert_info_position
 4:Delete_info_position
 5:Display_list
 6:Exit
Enter the choice : 4
Enter the position:
3
Invalid position

 1:Insert_rear
 2:Delete_rear
 3:Insert_info_position
 4:Delete_info_position
 5:Display_list
 6:Exit
Enter the choice : 45


------------------
(program exited with code: 0)

Press any key to continue . . .
```