/*merging of two single linked list*/

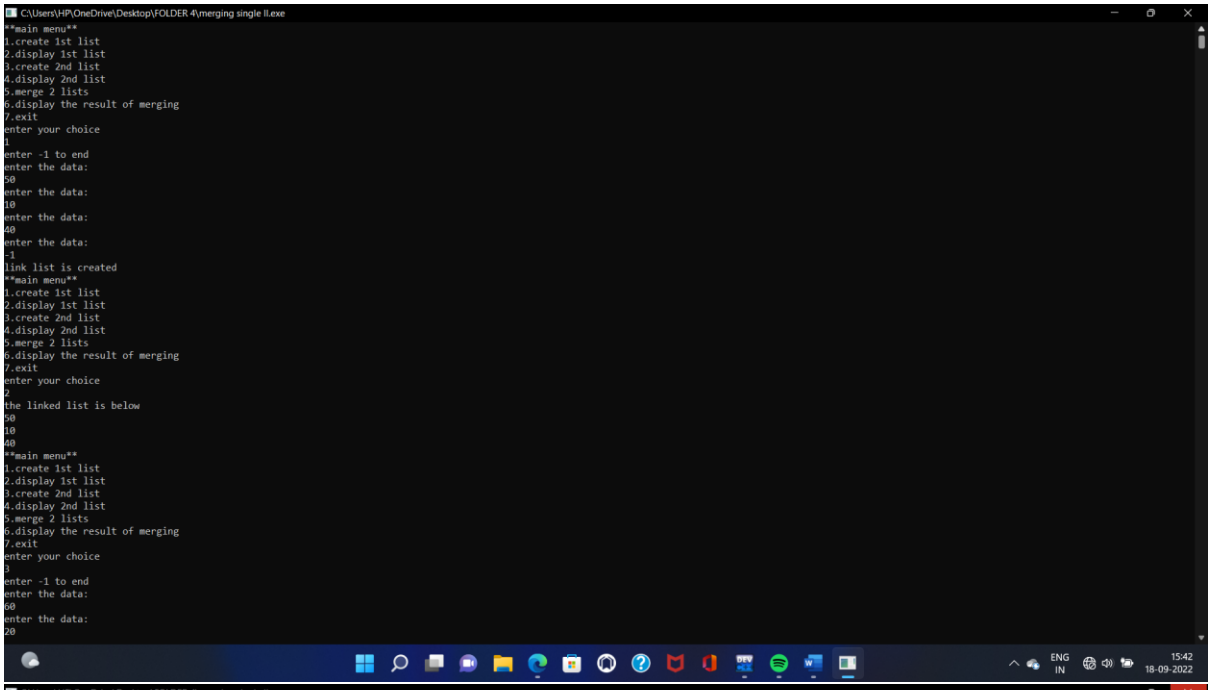
```
#include<stdio.h>
#include<stdlib.h>
struct node
{
	int data;
	struct node*link;
};
struct node*header1;
struct node*header2;
struct node*headermerge;
struct node*headercon;
struct node*create_ll(struct node*);
struct node*display(struct node*);
struct node*merging(struct node*);
struct node*concatination(struct node*,struct node*,struct node*);
int main()
{
	int choice=0;
	while(choice!=7)
	{
		printf("**main menu**\n");
		printf("1.create 1st list\n2.display 1st list\n3.create 2nd list\n4.display 2nd
list\n5.merge 2 lists\n6.display the result of merging\n7.exit\n");
		printf("enter your choice\n");
		scanf("%d",&choice);
		switch(choice)
		{
			case 1:header1=create_ll(header1);
			break;
			case 2:header1=display(header1);
```

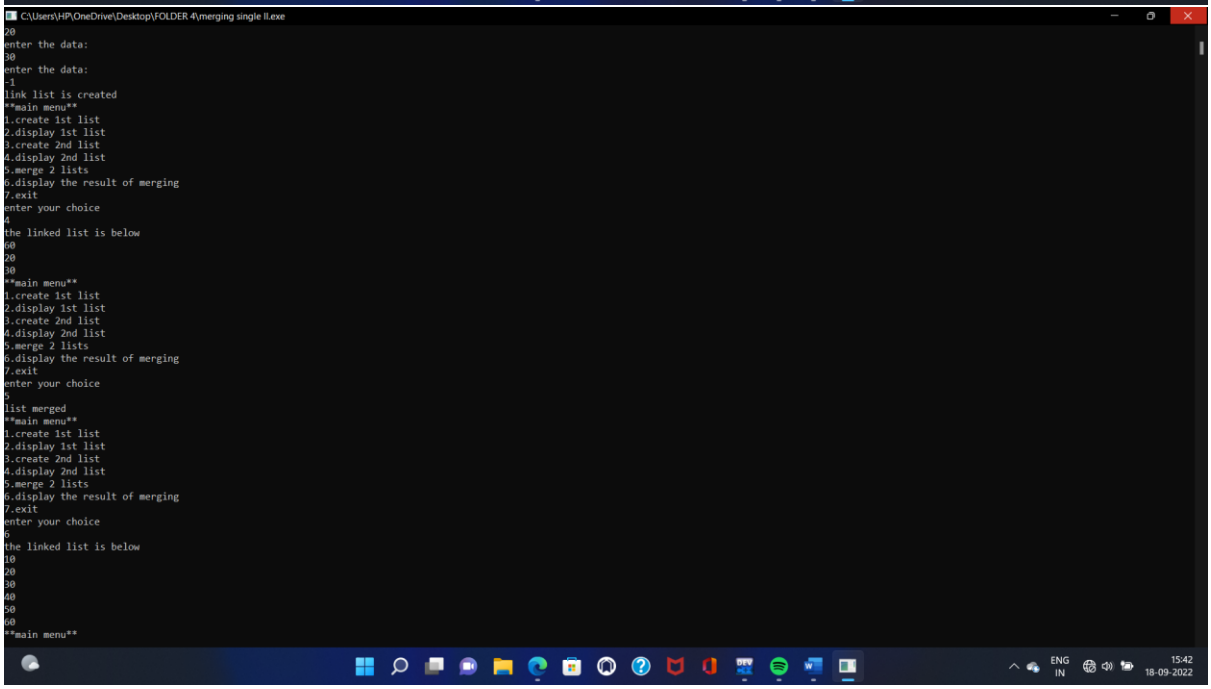
```
                        break;
                        case 3:header2=create_ll(header2);
                        break;
                        case 4:header2=display(header2);
                        break;
                        case 5:headermerge=merging(headermerge);
                        break;
                        case 6:headermerge=display(headermerge);
                        break;
                        case 7:exit(0);
                        default:
                                printf("invalid choice\n");
            }
   }
}
struct node*create_ll(struct node*header)
{
        struct node*new_node,*ptr;
        int item;
        printf("enter -1 to end\n");
        printf("enter the data: \n");
        scanf("%d",&item);
        while(item!=-1)
        {
                new_node=(struct node*)malloc(sizeof(struct node*));
                new_node->data=item;
                if(header==NULL)       //list is empty
                {
                        new_node->link=NULL;
                        header=new_node;
                }
```

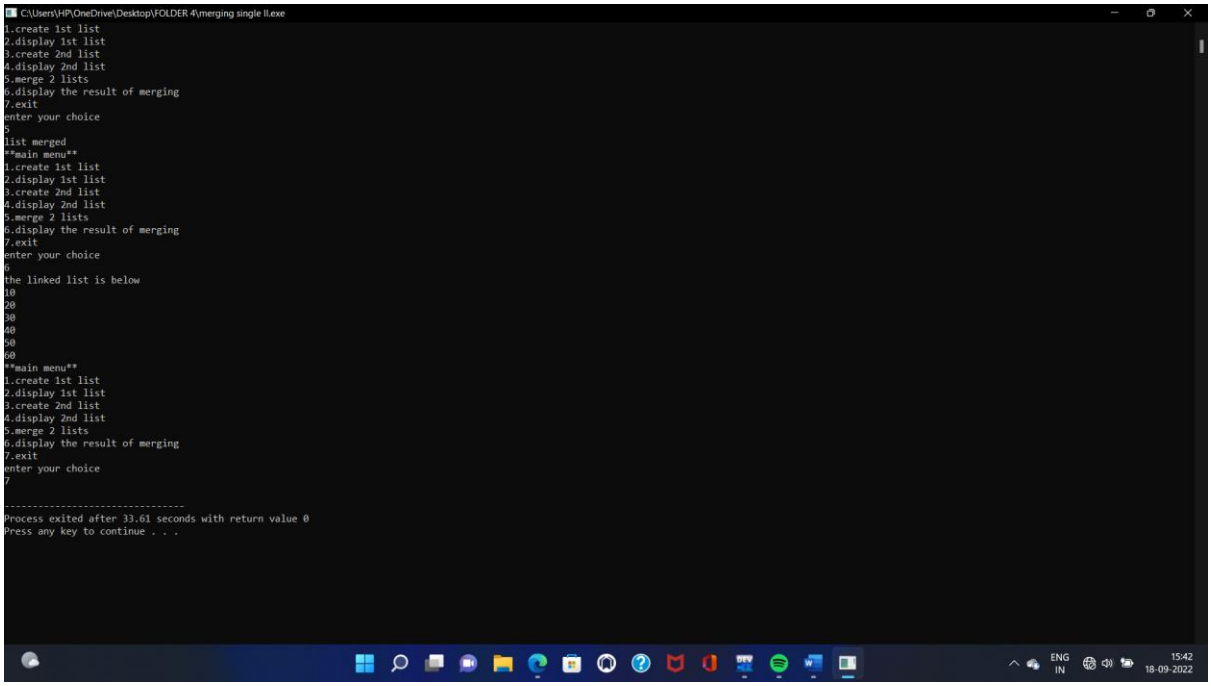
```
                else
                {
                        ptr=header;
                        while(ptr->link!=NULL)
                        {
                                ptr=ptr->link;
                   }
                                ptr->link=new_node;
                                new_node->link=NULL;
                }
                printf("enter the data: \n");
                scanf("%d",&item);
        }
        printf("link list is created\n");
        return header;
}
struct node*display(struct node*header)
{
        printf("the linked list is below\n");
        struct node*ptr;
        ptr=header;
        while(ptr!=NULL)        //list is not empty
        {
                printf("%d\n",ptr->data);
                ptr=ptr->link;
   }
   return header;
}
struct node*merging(struct node*headermerge)
{
        struct node*ptr1,*ptr2;
```

```
	int temp;
	headermerge=concatination(header1,header2,headercon);
	ptr1=headermerge;
	while(ptr1->link!=NULL)
	{
		ptr2=ptr1->link;
		while(ptr2!=NULL)	//there are atleast 2 nodes in the list
		{
			if(ptr1->data>ptr2->data)
			{
				temp=ptr1->data;
				ptr1->data=ptr2->data;
				ptr2->data=temp;
			}
			ptr2=ptr2->link;
		}
		ptr1=ptr1->link;
	}
	printf("list merged\n");
	return headermerge;
}
struct node*concatination(struct node*header1,struct node*header2,struct node*headercon)
{
	struct node*ptr;
	ptr=header1;
	while(ptr->link!=NULL)
	{
		ptr=ptr->link;
	}
	ptr->link=header2;
	headercon=header1;
```

```
        return headercon;
}
```


C:\Users\HP\OneDrive\Desktop\FOLDER 4\merging single ll.exe
main menu
1.create 1st list
2.display 1st list
3.create 2nd list
4.display 2nd list
5.merge 2 lists
6.display the result of merging
7.exit
enter your choice
1
enter -1 to end
enter the data:
50
enter the data:
10
enter the data:
40
enter the data:
-1
link list is created
main menu
1.create 1st list
2.display 1st list
3.create 2nd list
4.display 2nd list
5.merge 2 lists
6.display the result of merging
7.exit
enter your choice
2
the linked list is below
50
10
40
main menu
1.create 1st list
2.display 1st list
3.create 2nd list
4.display 2nd list
5.merge 2 lists
6.display the result of merging
7.exit
enter your choice
3
enter -1 to end
enter the data:
60
enter the data:
20
ENG
IN
15:42
18-09-2022


C:\Users\HP\OneDrive\Desktop\FOLDER 4\merging single ll.exe
20
enter the data:
30
enter the data:
-1
link list is created
main menu
1.create 1st list
2.display 1st list
3.create 2nd list
4.display 2nd list
5.merge 2 lists
6.display the result of merging
7.exit
enter your choice
4
the linked list is below
60
20
30
main menu
1.create 1st list
2.display 1st list
3.create 2nd list
4.display 2nd list
5.merge 2 lists
6.display the result of merging
7.exit
enter your choice
5
list merged
main menu
1.create 1st list
2.display 1st list
3.create 2nd list
4.display 2nd list
5.merge 2 lists
6.display the result of merging
7.exit
enter your choice
6
the linked list is below
10
20
30
40
50
60
main menu
ENG
IN
15:42
18-09-2022


C:\Users\HP\OneDrive\Desktop\FOLDER 4\merging single ll.exe
1.create 1st list
2.display 1st list
3.create 2nd list
4.display 2nd list
5.merge 2 lists
6.display the result of merging
7.exit
enter your choice
5
list merged
main menu
1.create 1st list
2.display 1st list
3.create 2nd list
4.display 2nd list
5.merge 2 lists
6.display the result of merging
7.exit
enter your choice
6
the linked list is below
10
20
30
40
50
60
main menu
1.create 1st list
2.display 1st list
3.create 2nd list
4.display 2nd list
5.merge 2 lists
6.display the result of merging
7.exit
enter your choice
7

Process exited after 33.61 seconds with return value 0
Press any key to continue . . .
ENG
IN
15:42
18-09-2022