Lab 6- Sorting and Reverse Concatenation

```
#include <stdio.h>
#include <stdlib.h>
struct node{
   int data;
   struct node *next;
};
struct node* head = NULL;
struct node* head2 = NULL;
void insert1(int d){
   struct node* new_node = (struct node*) malloc(sizeof(struct node));
   new_node->data = d;
   new_node->next=head;
   head=new_node;

}
void insert2(int d){
   struct node* new_node = (struct node*) malloc(sizeof(struct node));
   new_node->data = d;
   new_node->next=head2;
   head2=new_node;

}
void delete(){
   if(head==NULL){
      printf("list is empty");
   }
   struct node* temp = head;
   head = temp->next;
   free(temp);
}
```

```
void sort(){
   struct node* curr=head, *index=NULL;
   int temp;
   if(head==NULL){
      return 0;
   }
   else{
      while(curr!=NULL){
         index=curr->next;
         while(index!=NULL){
            if(curr->data > index->data){
               temp = index->data;
               index->data=curr->data;
               curr->data=temp;
            }
            index=index->next;
         }
         curr=curr->next;
      }
   }
}
struct node* reverse(){
   struct node *prev,*curr,*front;void display(){
   struct node* temp=head;
   while(temp!=NULL){
      printf("%d\t",temp->data);
      temp = temp->next;
   }
   printf("\n");
}
```

```
    prev = NULL;
   front=NULL;
    curr = head;
    while(curr!=NULL){
        front = curr->next;
        curr->next = prev;
        prev = curr;
        curr = front;
    }
    return prev;
}

void concatenate(){
    struct node* temp = head;
    while(temp->next!=NULL){
        temp = temp->next;
    }
    temp->next=head2;
}
void display1(){
    struct node* temp=head;
    while(temp!=NULL){
        printf("%d\t",temp->data);
        temp = temp->next;
    }
    printf("\n");
}
void display2(){
    struct node* temp=head2;
    while(temp!=NULL){
        printf("%d\t",temp->data);
```

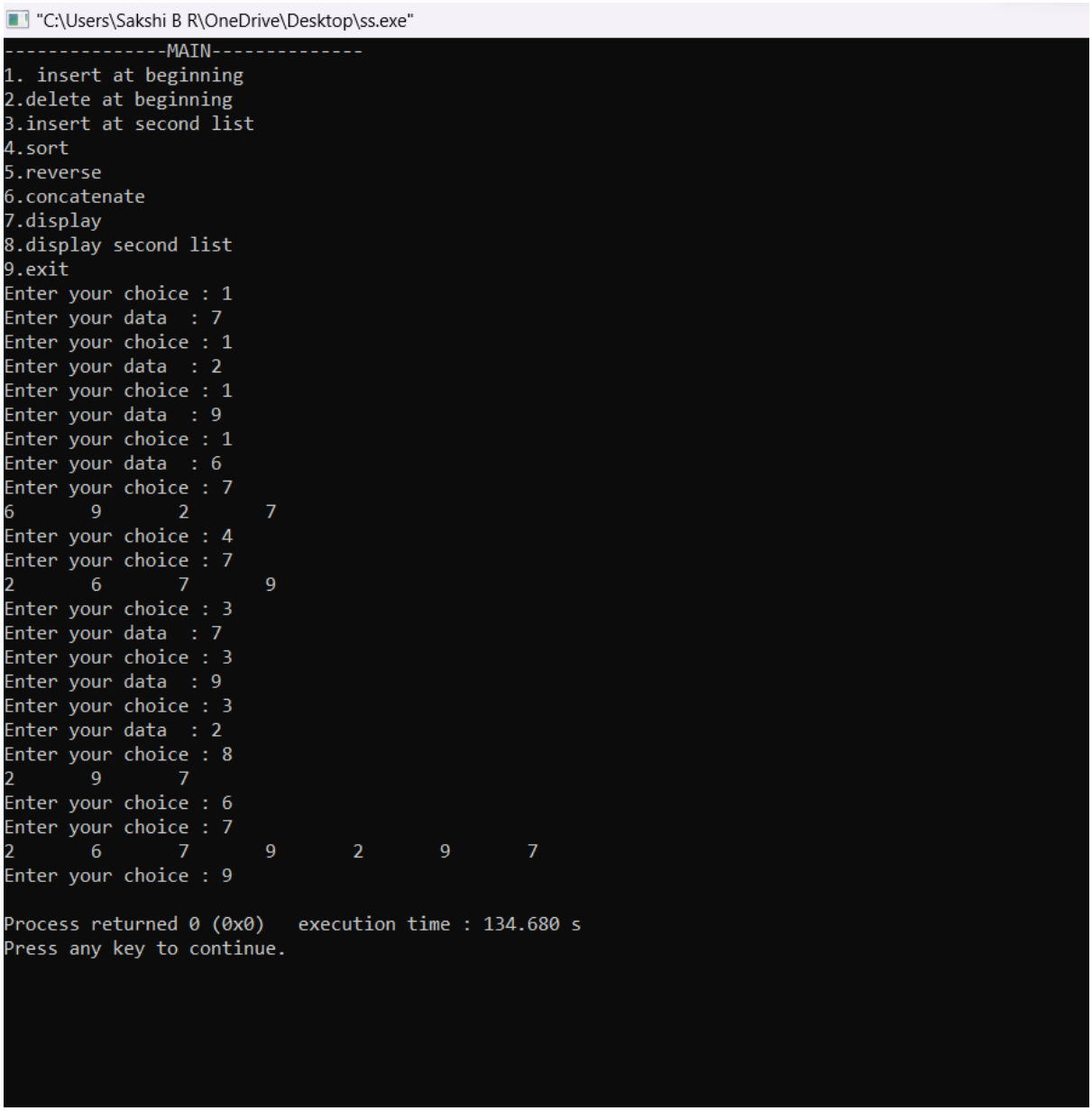
```
        temp = temp->next;
    }
    printf("\n");
}
int main(){

int ch,d;
printf("----------MAIN----------\n");
printf("1. insert at beginning \n2.delete at beginning\n3.insert at second list\n4.sort \n5.reverse \n6.concatenate\n7.display \n8.display second list \n9.exit\n");
while(ch!=9){
    printf("Enter your choice : ");
    scanf("%d",&ch);
    switch(ch){
     case 1: printf("Enter your data  : ");
            scanf("%d",&d);
            insert1(d);break;
     case 2: delete();break;
     case 3: printf("Enter your data  : ");
            scanf("%d",&d);
            insert2(d);break;
     case 4: sort();break;
     case 5: head=reverse();break;
     case 6: concatenate();break;
     case 7: display1(); break;
     case 8: display2(); break;
     case 9: exit(0);
     default : printf("Invalid number");

    }
}
```

return 0;

}


"C:\Users\Sakshi B R\OneDrive\Desktop\ss.exe"
--------------MAIN--------------
1. insert at beginning
2.delete at beginning
3.insert at second list
4.sort
5.reverse
6.concatenate
7.display
8.display second list
9.exit
Enter your choice : 1
Enter your data : 7
Enter your choice : 1
Enter your data : 2
Enter your choice : 1
Enter your data : 9
Enter your choice : 1
Enter your data : 6
Enter your choice : 7
6 9 2 7
Enter your choice : 4
Enter your choice : 7
2 6 7 9
Enter your choice : 3
Enter your data : 7
Enter your choice : 3
Enter your data : 9
Enter your choice : 3
Enter your data : 2
Enter your choice : 8
2 9 7
Enter your choice : 6
Enter your choice : 7
2 6 7 9 2 9 7
Enter your choice : 9
Process returned 0 (0x0) execution time : 134.680 s
Press any key to continue.