LAB -3

PROGRAM – SINGLY LINKED LIST

SOURCE CODE:

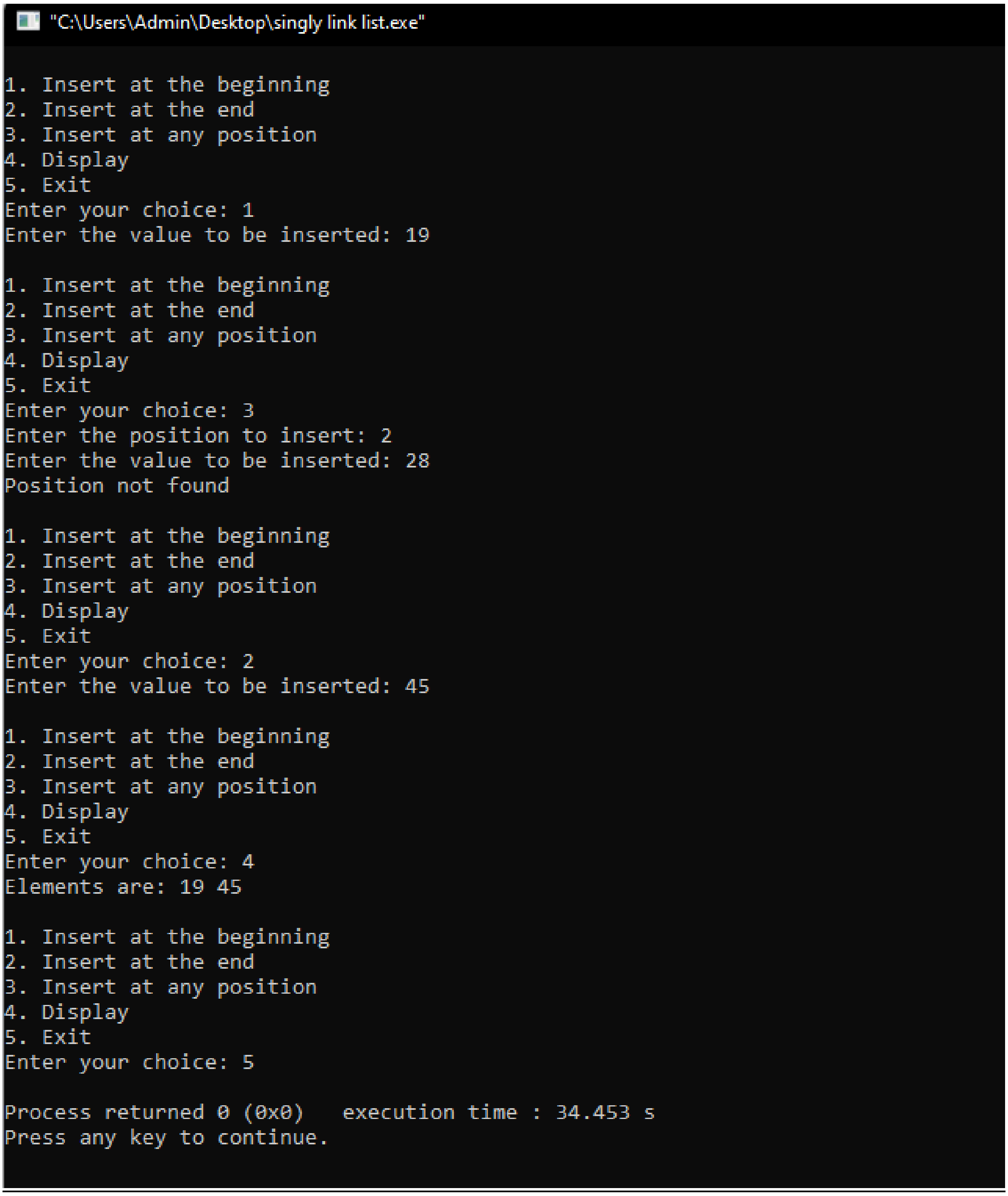
```
#include<stdio.h>
#include<stdlib.h>
struct node {
    int data;
    struct node *next;
};
struct node *head = NULL;
void display() {
    struct node *ptr = head;
    if (ptr == NULL) {
        printf("List is empty\n");
        return;
    }
    printf("Elements are: ");
    while (ptr != NULL) {
        printf("%d ", ptr->data);
        ptr = ptr->next;
    }
    printf("\n");
}
void insert_begin() {
    struct node *temp;
    temp = (struct node *)malloc(sizeof(struct node));
    printf("Enter the value to be inserted: ");
    scanf("%d", &temp->data);
    temp->next = head;
    head = temp;
}
void insert_end() {
    struct node *temp, *ptr;
    temp = (struct node *)malloc(sizeof(struct node));
    printf("Enter the value to be inserted: ");
    scanf("%d", &temp->data);
    temp->next = NULL;
    if (head == NULL) {
        head = temp;
    } else {
        ptr = head;
        while (ptr->next != NULL) {
            ptr = ptr->next;
        }
        ptr->next = temp;
    }
```

```
}
void insert_pos() {
    int pos, i;
    struct node *temp, *ptr;
    temp = (struct node *)malloc(sizeof(struct node));
    printf("Enter the position to insert: ");
    scanf("%d", &pos);
    printf("Enter the value to be inserted: ");
    scanf("%d", &temp->data);
    temp->next = NULL;

    if (pos == 0) {
        temp->next = head;
        head = temp;
    } else {
        ptr = head;
        for (i = 0; i < pos - 1; i++) {
            ptr = ptr->next;
            if (ptr == NULL) {
                printf("Position not found\n");
                return;
            }
        }
        temp->next = ptr->next;
        ptr->next = temp;
    }
}
int main() {
    int choice;
    while(1) {
        printf("\n1. Insert at the beginning\n2. Insert at the end\n3. Insert at any position\n4.
Display\n5. Exit\n");
        printf("Enter your choice: ");
        scanf("%d", &choice);

        switch(choice) {
            case 1:
                insert_begin();
                break;
            case 2:
                insert_end();
                break;
            case 3:
                insert_pos();
                break;
            case 4:
                display();
                break;
        case 5:
          exit(0);
          break;
        default:
          printf("Enter the correct choice\n");
      }
    }
  return 0;
}
```

OUTPUT:



"C:\Users\Admin\Desktop\singly link list.exe"

```
1. Insert at the beginning
2. Insert at the end
3. Insert at any position
4. Display
5. Exit
Enter your choice: 1
Enter the value to be inserted: 19

1. Insert at the beginning
2. Insert at the end
3. Insert at any position
4. Display
5. Exit
Enter your choice: 3
Enter the position to insert: 2
Enter the value to be inserted: 28
Position not found

1. Insert at the beginning
2. Insert at the end
3. Insert at any position
4. Display
5. Exit
Enter your choice: 2
Enter the value to be inserted: 45

1. Insert at the beginning
2. Insert at the end
3. Insert at any position
4. Display
5. Exit
Enter your choice: 4
Elements are: 19 45

1. Insert at the beginning
2. Insert at the end
3. Insert at any position
4. Display
5. Exit
Enter your choice: 5

Process returned 0 (0x0)   execution time : 34.453 s
Press any key to continue.
```