

Explorer

SingleLL1... x unionSLL.c x

Submit

Debugger

Plots

```
1  struct node {
2      int data;
3      struct node *next;
4  };
5  typedef struct node *NODE;
6
7  NODE createAndAddNodes(NODE first) {
8      NODE temp, q;
9      int x;
10     printf("Enter element : ");
11     scanf("%d", &x);
12     while(x != -1) {
13         temp = (NODE) malloc(sizeof(struct node));
14         temp->data = x;
15         temp->next = NULL;
16         if(first == NULL) {
17             first = temp;
18         } else {
19             q->next = temp;
20         }
21         q = temp;
22         printf("Enter element : ");
23         scanf("%d", &x);
24     }
25     return first;
26 }
27
28 void print(NODE first) {
29     NODE q = first;
30     if(first == NULL) {
31         printf("Single Linked List is empty\n");
32     } else {
33         printf("Elements in the list are : ");
34         while(q != NULL) {
35             printf("%d---> ", q->data);
36             q = q->next;
37         }
38         printf("NULL\n");
39     }
40 }
41
42 NODE concatenate(NODE t1, NODE t2)
43 {
```

< Prev

Reset

Submit

Next >

```
45     NODE·temp·==·t1;
46     if(·t1·==·NULL·)
47     {
48         return·t2;
49     }
50     if(·t2·==·NULL·)
51     {
52         return·t1;
53     }
54     while(·temp->next·!=·NULL·)
55     {
56         temp·==·temp->next;
57     }
58
59     temp->next·==·t2;
60
61     return·t1;
62 }
63
64 NODE·unionSLL(NODE·l1,·NODE·l2)·
65 {
66     NODE·l3·==concatenate(l1,l2);
67
68     NODE·t1,t2,q;
69
70     for(·t1·==·l3·;·t1!=·NULL·&&·t1->next·!=·NULL·;·t1·==·t1->next·)
71     {
72         for(·t2·==·t1·;·t2·!=·NULL·&&·t2->next·!=·NULL·;·)
73         {
74             if(·t1->data·==·t2->next->data·)
75             {
76                 q·==·t2->next;
77                 t2->next·==·q->next;
78             }
79             else
80             {
81                 t2·==·t2->next;
82             }
83         }
84     }
85     return·l3;
86
87
88 }
89
```

