```
[] G & Share Run
                                                                               Output
                                                                                                                                            Clear
       main.c
       1 #include <stdio.h>
                                                                             1.Insert 2.Display 3.Delete 4.Exit
      2 #include <stdlib.h>
       3 - typedef struct Node {
                                                                             Choice: 2
             int data;
                                                                             List is empty.
              struct Node *next;
                                                                             1.Insert 2.Display 3.Delete 4.Exit
       6 } Node;
5
       7- void insert(Node **tail, int val) {
                                                                             Choice: 1
             Node *n = malloc(sizeof(Node));
                                                                             Enter value: 10
       8
              n->data = val;
       9
                                                                             1.Insert 2.Display 3.Delete 4.Exit
             if (!*tail) {
      10 -
0
                n->next = n;
                                                                             Choice:
       11
       12
                 *tail = n;
              } else {
      13 -
       14
                 n->next = (*tail)->next;
                 (*tail)->next = n;
      15
0
      16
                 *tail = n;
      17
      18 }
      19 - void delete(Node **tail, int val) {
             if (!*tail) return;
TS
      20
              Node *curr = (*tail)->next, *prev = *tail;
      21
      22 *
              do {
                 if (curr->data == val) {
      23 -
      24 -
                     if (curr == *tail && curr->next == *tail) {
      25
                         free(curr);
      26
                         *tail = NULL;
```

```
[] G & Share Run
ф
       27
                        return;
Q
                                                                           1.Insert 2.Display 3.Delete 4.Exit
      28
                                                                           Choice: 2
                     if (curr == *tail) *tail = prev;
       29
                                                                           List is empty.
1
                     prev->next = curr->next;
      30
       31
                     free(curr);
                                                                           1.Insert 2.Display 3.Delete 4.Exit
日
                    return;
                                                                           Choice: 1
       33
                                                                           Enter value: 10
                 prev = curr;
叁
       35
                 curr = curr->next;
                                                                           1.Insert 2.Display 3.Delete 4.Exit
       36
             } while (curr != (*tail)->next);
0
                                                                           Choice:
       37 }
       38 - void display(Node *tail) {
0
      39 -
             if (!tail) {
               printf("List is empty.\n");
       40
0
      41
                 return;
      42
             Node *p = tail->next;
      43
             printf("List: ");
       44
      45 -
             do {
             printf("%d -> ", p->data);
p = p->next;
} while (p != tail->next);
      46
      47
      48
             printf("(back to head)\n");
      49
      50 }
      51 - int main() {
52 Node *tail = NULL;
```

Output

main.c

Clear

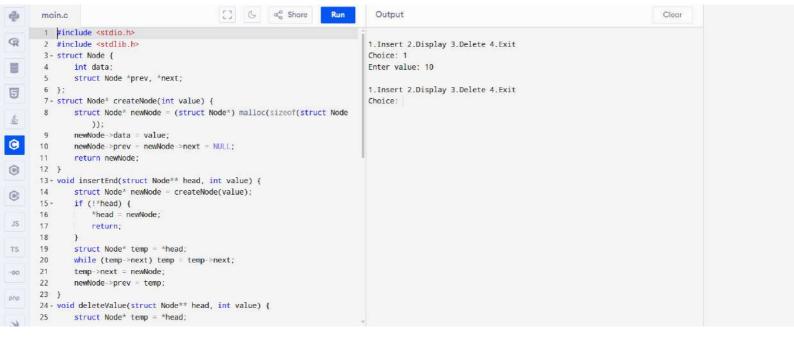
```
[] G Share Run
                                                                                 Output
right.
                                                                                                                                                    Clear
       45 -
               do {
Q
                                                                                  1.Insert 2.Display 3.Delete 4.Exit
                printf("%d -> ", p->data);
       46
                                                                                  Choice: 2
       47
                   p = p->next;
} while (p != tail->next);
                                                                                  List is empty.
       48
              printf("(back to head)\n");
       49
                                                                                  1.Insert 2.Display 3.Delete 4.Exit
5
       50 }
       51 - int main() {
52    Node *tail = NULL;
53    int ch, val;
                                                                                  Choice: 1
                                                                                  Enter value: 10
恋
                                                                                  1.Insert 2.Display 3.Delete 4.Exit
       54 -
               while (1) {
0
                 printf("\n1.Insert 2.Display 3.Delete 4.Exit\nChoice: ");
       55
                   scanf("%d", &ch);
       56
0
       57 -
                  if (ch == 1) {
                   printf("Enter value: ");
scanf("%d", &val);
insert(&tail, val);
       58
0
       59
       60
       61 -
                } else if (ch == 2) {
 JS
       62
                     display(tail);
       63 -
                  } else if (ch == 3) {
                     printf("Enter value to delete: ");
       65
                      scanf("%d", &val);
                      delete(&tail, val);
                  } else break;
       68
       69
               return 0;
       70 }
```

```
[] & & Share
                                                                                 Output
                                                                                                                                                Clear
                                                                      Run
       main.c
       1 #include <stdio.h>
       2 #include <stdlib.h>
                                                                                1.Insert 2.Display 3.Delete 4.Exit
       3 - typedef struct Node {
                                                                               Choice: 1
                                                                               Enter value: 10
             int data;
       4
              struct Node *prev, *next;
       6 } Node;
                                                                               1.Insert 2.Display 3.Delete 4.Exit
       7 - void insert(Node **h, int v) {
                                                                               Choice: 1
              Node *n = malloc(sizeof(Node)), *t = *h;
                                                                               Enter value: 20
       8
              n->data = v; n->next = NULL;
       9
       10
             if (!t) { n->prev = NULL; *h = n; return; }
                                                                               1.Insert 2.Display 3.Delete 4.Exit
0
             while (t->next) t = t->next;
       11
                                                                               Choice:
       12
              t->next = n; n->prev = t;
0
       13 }
       14 - void del(Node **h, int v) {
              Node *t = *h;
      15
              while (t && t->data != v) t = t->next;
       16
       17
             if (!t) return;
              if (t->prev) t->prev->next = t->next; else *h = t->next;
       18
       19
              if (t->next) t->next->prev = t->prev;
      20
              free(t);
TS
      21 }
      22 - void show(Node *t) {
      23
              printf("List: ");
              while (t) { printf("%d <-> ", t->data); t = t->next; }
      24
      25
              printf("NULL\n");
      26 }
```

9

0

```
🗀 🕓 🖒 Share Run
                                                                                       Output
                                                                                                                                                           Clear
        main.c
        21 }
Q
                                                                                      1.Insert 2.Display 3.Delete 4.Exit
        22 - void show(Node *t) {
                                                                                      Choice: 1
        23
               printf("List: ");
                                                                                      Enter value: 10
H
                while (t) { printf("%d <-> ", t->data); t = t->next; }
        25
               printf("NULL\n");
       26 }
                                                                                      1.Insert 2.Display 3.Delete 4.Exit
9
                                                                                      Choice: 1
        27 - int main() {
                                                                                      Enter value: 20
               Node *head = NULL;
        28
盐
        29
                int ch, val;
                while (1) {
                                                                                      1.Insert 2.Display 3.Delete 4.Exit
        30 -
0
                    printf("\n1.Insert 2.Display 3.Delete 4.Exit\nChoice: ");
                                                                                      Choice:
        31
        32
                    scanf("%d", &ch);
                   scant("%d", odi),
if (ch = 1) {
    printf("Enter value: ");
    scanf("%d", &val);
}
0
        33 -
        34
        35
0
        36
                       insert(&head, val);
        37 -
                   } else if (ch == 2) {
 JS
                    show(head);
} else if (ch == 3) {
        38
        39 -
 TS
                       printf("Enter value to delete: ");
        40
        41
                        scanf("%d", &val);
       42
                        del(&head, val);
        43
                    } else break:
       44
       45
                return 0;
     46 }
```



```
[] & a Share Run
       main.c
                                                                                 Output
                                                                                                                                                 Clear
              while (temp && temp->data != value) temp = temp->next;
       26
                                                                                1.Insert 2.Display 3.Delete 4.Exit
       27
               if (!temp) return;
                                                                                Choice: 1
       28
               if (temp->prev) temp->prev->next = temp->next;
H
                                                                                Enter value: 10
       29
               else *head = temp->next;
       30
               if (temp->next) temp->next->prev = temp->prev;
                                                                                1.Insert 2.Display 3.Delete 4.Exit
       31
               free(temp):
O
                                                                                Choice:
       33 - void display(struct Node* head) {
$
       34 -
              while (head) {
       35
                  printf("%d <-> ". head->data);
0
                  head = head->next;
       36
       37
0
              printf("NULL\n");
       38
       39 }
       40 - int main() {
0
              struct Node* head = NULL;
       41
       42
              int ch, val:
              while (1) {
    printf("\n1.Insert 2.Display 3.Delete 4.Exit\nChoice: ");
       43 -
       44
TS
       45
                  scanf("%d", &ch);
                  if (ch == 1) {
       46 -
       47
                     printf("Enter value: ");
                      scanf("%d", &val);
       48
                      insertEnd(&head, val);
       49
       50 -
                  } else if (ch == 2) {
                     display(head);
1
```

```
[] G of Share Run
       main.c
                                                                                   Output
                                                                                                                                                     Clear
è
       34 -
               while (head) {
                                                                                  1.Insert 2.Display 3.Delete 4.Exit
Q
                   printf("%d <-> ", head->data);
       35
                                                                                  Choice: 1
       36
                   head = head->next;
Enter value: 10
       37
       38
               printf("NULL\n");
                                                                                  1.Insert 2.Display 3.Delete 4.Exit
       39 }
回
                                                                                  Choice:
       40 - int main() {
       41
               struct Node* head = NULL;
$
               int ch, val;
       42
       43 -
               while (1) {
0
                  printf("\n1.Insert 2.Display 3.Delete 4.Exit\nChoice: ");
       44
                  scanf("%d", &ch);
if (ch == 1) {
       45
0
       46 -
                   printf("Enter value: ");
scanf("%d", &val);
       47
       48
0
                 insertEnd(&head, val);
} else if (ch = 2) {
       49
       50 -
 JS
       51
                      display(head);
                  } else if (ch == 3) {
       52 -
 TS
       53
                     printf("Enter value to delete: ");
                       scanf("%d", &val):
       54
       55
                      deleteValue(&head, val);
       56
                   } else break;
       57
       58
               return 0;
34
```