

```

1: // A simple and tail recursive C program to reverse a linked
2: #include <stdio.h>
3: #include <stdlib.h>
4:
5: typedef struct Node {
6:     int data;
7:     struct Node* next;
8: }Node;
9:
10: void reverseUtil(Node* curr, Node* prev, Node** head);
11:
12: // This function mainly calls reverseUtil()
13: // with prev as NULL
14: void reverse(Node** head)
15: {
16:     if (!head)
17:         return;
18:     reverseUtil(*head, NULL, head);
19: }
20:
21: // A simple and tail-recursive function to reverse
22: // a linked list. prev is passed as NULL initially.
23: void reverseUtil(Node* curr, Node* prev, Node** head)
24: {
25:     /* If last node mark it head*/
26:     if (!curr->next) {
27:         *head = curr;
28:         /* Update next to prev node */
29:         curr->next = prev;
30:         return;
31:     }
32:
33:     /* Save curr->next node for recursive call */
34:     Node* next = curr->next;
35:     /* and update next */
36:     curr->next = prev;
37:     reverseUtil(next, curr, head);
38: }
39:

```

```

40: // A utility function to create a new node
41: Node* newNode(int key)
42: {
43:     Node* temp = (Node *)malloc(sizeof(Node));
44:     temp->data = key;
45:     temp->next = NULL;
46:     return temp;
47: }
48:
49: // A utility function to print a linked list
50: void printlist(Node* head)
51: {
52:     while (head != NULL) {
53:         printf("%d ", head->data);
54:         head = head->next;
55:     }
56:     printf("\n");
57: }
58:
59: // Driver code
60: int main()
61: {
62:     Node* head1 = newNode(1);
63:     head1->next = newNode(2);
64:     head1->next->next = newNode(3);
65:     head1->next->next->next = newNode(4);
66:     head1->next->next->next->next = newNode(5);
67:     head1->next->next->next->next->next = newNode(6);
68:     head1->next->next->next->next->next->next = newNode(7);
69:     head1->next->next->next->next->next->next->next = newNode(8);
70:     printf("Given linked list\n");
71:     printlist(head1);
72:     reverse(&head1);
73:     printf("\nReversed linked list\n");
74:     printlist(head1);
75:     return 0;
76: }
77:
78: // This code is contributed by Aditya Kumar (adityakumar129)

```

79: