Afnan-15cs058
20-may-2020

If a
linked
listis:
1 → 2 →
3 → 4 →
5 → 6 →
7 → 8

The value of size k is 2

Then the linked list looks like: 2 → 1 → 4 → 3 → 6 → 5 → 8 → 7

If a linked listis: 1 → 2 → 3 → 4 → 5 → 6 → 7 → 8

The value of size k is 3

Then the linked list looks like: 3 → 2 → 1 → 6 → 5 → 4 → 8 → 7

```
       struct Node
       {
       int data;
       struct Node* next;
       };
       pointer to the new head node. /
       struct Node reverse (struct Node head, int k)
       {
       struct Node current = head;
       struct Node next = NULL;
       struct Node prev = NULL;
       int count = 0;
       while (current != NULL && count < k) { next = current->next; current-
>next = prev; prev = current; current = next; count++; } if (next != NULL)
head->next = reverse(next, k); return prev;
       }
       void push(struct Node** head_ref, int new_data)
       {
       struct Node* new_node =
       (struct Node*) malloc(sizeof(struct Node));
       new_node->data = new_data; new_node->next = (*head_ref); (*head_ref) =
new_node;
       }
       void printList(struct Node *node)
       {
       while (node != NULL)
       {
```

```
       printf("%d ", node->data);
       node = node->next;
       }
       }
       int main(void)
       {
       struct Node* head = NULL;
       push(&head, 8);
       push(&head, 7);
       push(&head, 6);
       push(&head, 5);
       push(&head, 4);
       push(&head, 3);
       push(&head, 2);
       push(&head, 1);
       printf("\nGiven linked list \n"); printList(head); head =
reverse(head, 2);
       printf("\nReversed Linked list \n"); printList(head); return(0);
       }
```