



PES
UNIVERSITY

CELEBRATING 50 YEARS

Data Structures

Dilip Kumar Maripuri
Computer Applications



Data Structures

Session : Applications of Linked Lists: Reverse a List, Sort a List

Dilip Kumar Maripuri
Computer Applications





Data Structures

Operations on Lists

► Reverse a given Singly Linked List

Algorithm Reverse_List(head):

1. Initialize $new_head \leftarrow NULL$
2. **while** head $\neq NULL$ **do**
 - 2.1 Set $data \leftarrow head.data$
 - 2.2 Set $head \leftarrow Delete_Front(head)$
 - 2.3 Set $new_head \leftarrow Insert_Front(new_head, data)$
3. **end while**
4. **return** new_head

End Algorithm

```
1  NODE reverse_list(NODE Head) {  
2      NODE new_head = NULL;  
3      int data;  
4      while (Head != NULL)  
5      {  
6          data = Head->data;  
7          Head = del_front(Head);  
8          new_head = ins_front(new_head, data);  
9      }  
10     return new_head;  
11 }
```



Algorithm Sort_List(head):

2. **while** head \neq NULL **do**

2.2 Set $head \leftarrow \text{Delete_Front}(head)$

2.3 Set *sorted_head* \leftarrow Insert_Order(*sorted_head*, *data*)

3. end while

4. **return** sorted_head

End Algorithm

```
1  NODE sort_list(NODE Head) {  
2      NODE sorted_head = NULL;  
3      int data;  
4  
5      while (Head != NULL) {  
6          int data = Head->data;  
7          Head = del_front(Head);  
8          sorted_head = insert_order(sorted_head,  
9                                     data);  
10     }  
11     return sorted_head;  
12 }
```



Thank You

Dilip Kumar Maripuri
Associate Professor
Department of Computer Applications
dilip.maripuri@pes.edu
8073212026

