

```

C 3_1_singly_linked_list.c > main()
1  /* Write a program in C to create and display a singly linked list. */
2
3  #include <stdio.h>
4  #include <stdlib.h>
5
6  struct node{
7      int data;
8      struct node *next;
9  };
10
11 int main(){
12     struct node *head = (struct node*)malloc(sizeof(struct node));
13     struct node *second = (struct node*)malloc(sizeof(struct node));
14     struct node *end = (struct node*)malloc(sizeof(struct node));
15     head->data=1;
16     head->next=second;
17     second->data=2;
18     second->next=end;
19     end->data=3;
20     end->next=NULL;
21     struct node *temp=head;
22     while(temp!=NULL){
23         printf("%d || %u, ",temp->data,temp->next);
24         temp=temp->next;
25     }
26     return 0;
27 }
28

```

TERMINAL COMMENTS

```

PS C:\Users\shuvr\OneDrive\Documents\College C codes\DSA-ASS-3> ^C
PS C:\Users\shuvr\OneDrive\Documents\College C codes\DSA-ASS-3>
PS C:\Users\shuvr\OneDrive\Documents\College C codes\DSA-ASS-3> & 'c:\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-co5wfnjx.41s'
r22f.wcg' '--pid=Microsoft-MIEngine-Pid-qsyflz3x.yf5' '--dbgExe=C:\msys
1 || 957504, 2 || 957568, 3 || 0,
PS C:\Users\shuvr\OneDrive\Documents\College C codes\DSA-ASS-3>

```

C 3_2_insertatbeginning.c > ...

```
1  /* Write a program in C to insert a new node at the beginning of a
2  Singly Linked List.*/
3
4  #include <stdio.h>
5  #include <stdlib.h>
6
7  struct node{
8      int data;
9      struct node *next;
10 };
11
12 struct node *createnode(int value){
13     struct node *newnode=(struct node*)malloc(sizeof(struct node));
14     newnode->data=value;
15     newnode->next=NULL;
16     return newnode;
17 }
18
19 void displaylist(struct node *head){
20     struct node *temp=head;
21     while(temp!=NULL){
22         printf("%d  ",temp->data);
23         temp=temp->next;
24     }
25 }
26
27 void insertatbeg(struct node **head,int value){
28     struct node *newnode=createnode(value);
29     newnode->next=*head;
30     *head=newnode;
31 }
32
33 int main(){
34     struct node *head=createnode(1);
35     struct node *second=createnode(2);
36     struct node *end=createnode(3);
37     head->next=second;
38     second->next=end;
39     end->next=NULL;
40     printf("List Before Insertion : ");
41     displaylist(head);
42     insertatbeg(&head,0);
43     insertatbeg(&head,-1);
44     printf("\nList After Insertion at beginning : ");
45     displaylist(head);
46     return 0;
47 }
```

List Before Insertion : 1 2 3

List After Insertion at beginning : -1 0 1 2 3

C 3_3_traverse_linked_list.c > ...

```
1  /* Write a program in C to traverse in a singly linked list.*/
2
3  #include <stdio.h>
4  #include <stdlib.h>
5
6  struct node{
7      int data;
8      struct node *next;
9  };
10
11 struct node *createnode(int value){
12     struct node *newnode=(struct node*)malloc(sizeof(struct node));
13     newnode->data=value;
14     newnode->next=NULL;
15     return newnode;
16 }
17
18 void traverselist(struct node *head){
19     struct node *temp=head;
20     while(temp!=NULL){
21         printf("%d | %u ,",temp->data,temp->next);
22         temp=temp->next;
23     }
24 }
25
26 int main(){
27     struct node *head=createnode(1);
28     struct node *second=createnode(2);
29     struct node *end=createnode(3);
30     head->next=second;
31     second->next=end;
32     end->next=NULL;
33     printf("Traversing the linked list...\nNodes of the linked list : ");
34     traverselist(head);
35     return 0;
36 }
```

TERMINAL COMMENTS

Traversing the linked list...
Nodes of the linked list : 1 | 7248976 ,2 | 7249040 ,3 | 0 ,

C 3_4_linkedlist_copy_arr.c > ...

```
1  /* Write a program in C to copy the elements of the array to a singly
2  linked list.*/
3
4  #include <stdio.h>
5  #include <stdlib.h>
6
7  struct node{
8      int data;
9      struct node *next;
10 };
11
12 struct node *createnode(int value){
13     struct node *newnode=(struct node*)malloc(sizeof(struct node));
14     newnode->data=value;
15     newnode->next=NULL;
16     return newnode;
17 }
18
19 void displaylist(struct node *head){
20     struct node *temp=head;
21     while(temp!=NULL){
22         printf("%d || %u, ",temp->data,temp->next);
23         temp=temp->next;
24     }
25 }
26
27 struct node *copy(int arr[],int length){
```

```

27 struct node *copy(int arr[],int length){
28     if(length==0){
29         return NULL;
30     }
31     struct node *head=createnode(arr[0]);
32     struct node *current=head;
33     for(int i=1;i<length;i++){
34         current->next=createnode(arr[i]);
35         current=current->next;
36     }
37     current->next=NULL;
38     return head;
39 }
40
41
42 int main(){
43     int length;
44     printf("Enter length of array : ");
45     scanf("%d",&length);
46     int arr[length];
47     printf("Enter %d numbers : ",length);
48     for(int i=0;i<length;i++){
49         scanf("%d",&arr[i]);
50     }
51     struct node *head=copy(arr,length);
52     printf("\nLinked List as a copy of the given array :--\n ");
53     displaylist(head);
54     return 0;
55 }

```

TERMINAL COMMENTS

Enter 5 numbers : 1 2 3 4 5

Linked List as a copy of the given array :--

1 || 7380032, 2 || 7380096, 3 || 7380160, 4 || 7380224, 5 || 0,

C 3_5_linkedlist_to_array.c > ...

```
1  /* Write a C program that converts a singly linked list into an array and
2  returns it */
3
4  #include <stdio.h>
5  #include <stdlib.h>
6
7  struct node{
8      int data;
9      struct node *next;
10 };
11
12 struct node *createnode(int value){
13     struct node *newnode=(struct node*)malloc(sizeof(struct node));
14     newnode->data=value;
15     newnode->next=NULL;
16     return newnode;
17 }
18
19 void displaylist(struct node *head){
20     struct node *temp=head;
21     while(temp!=NULL){
22         printf("%d || %u, ",temp->data,temp->next);
23         temp=temp->next;
24     }
25 }
26
27 int listlength(struct node *head){
28     struct node *temp=head;
29     int c=0;
30     while(temp!=NULL){
31         c++;
32         temp=temp->next;
33     }
34     return c;
35 }
36
```

```

36
37 int *convert(struct node *head,int length){
38     int *arr=(int *)malloc(length*(sizeof(int)));
39     int index=0;
40     struct node *temp=head;
41     while(temp!=NULL){
42         arr[index]=temp->data;
43         temp=temp->next;
44         index++;
45     }
46     return arr;
47 }
48
49 int main(){
50     // Creating sample linked list
51     struct node *head=createnode(1);
52     struct node *second=createnode(2);
53     struct node *third=createnode(3);
54     struct node *fourth=createnode(4);
55     struct node *end=createnode(5);
56     head->next=second;
57     second->next=third;
58     third->next=fourth;
59     fourth->next=end;
60     end->next=NULL;
61     printf("Linked List :--\n");
62     displaylist(head);
63     int length=listlength(head);
64     int *arr=convert(head,length);
65     printf("\nConverted Array :--\n");
66     for(int i=0;i<length;i++){
67         printf("%d  ",*(arr+i));
68     }
69     return 0;
70 }

```

TERMINAL COMMENTS

Linked List :--

1 || 6921280, 2 || 6921344, 3 || 6921408, 4 || 6921472, 5 || 0,

Converted Array :--

1 2 3 4 5