

# WEEK 7

```
#include <stdio.h>

#include <stdlib.h>

struct node
{
    int data;
    struct node *next;
};

struct node *head=NULL;
struct node *head1=NULL;
struct node *head2=NULL;
struct node *head3=NULL;

void Reverse()
{
    struct node *newnode,*temp;
    int item;
    int choice;
    do
    {
        newnode =(struct node *) malloc (sizeof(struct node));
        printf("Enter the data : ");
        scanf("%d",&item);
        newnode->data=item;
        newnode->next=NULL;
```

```
if (head==NULL)
{

    head=newnode;

}
else
{
temp=head;
    while(temp->next!=NULL)
    {
        temp=temp->next;
    }
temp->next=newnode;
newnode->next=NULL;

}

printf("Do u want to add element 1-yes, 2-no\n");
fflush(stdin);
scanf("%d",&choice);
}while(choice==1);


struct node *prev=NULL,*current=head, *next=NULL;
while(current!=NULL)
{
    next=current->next;
```

```

        current->next=prev;
        prev=current;
        current=next;
    }
    head=prev;

    printf("DISPLAY:\n");
    struct node *ptr;
    ptr=head;

    if(ptr==NULL)
    {
        printf("Nothing to print\n");
    }
    else
    {
        while(ptr!=NULL)
        {
            printf("%d ",ptr->data);
            ptr=ptr->next;
        }
    }
}

void Concat()
{
    struct node *newnode1,*temp1;

```

```
int item1;
printf("LIST ONE ELEMENTS\n");
int choice1;
do
{
newnode1 =(struct node *) malloc (sizeof(struct node));
printf("Enter the data : ");
scanf("%d",&item1);
newnode1->data=item1;
newnode1->next=NULL;
if (head1==NULL)
{

head1=newnode1;

}
else
{
temp1=head1;
while(temp1->next!=NULL)
{
temp1=temp1->next;
}
temp1->next=newnode1;
newnode1->next=NULL;
```

```

}

printf("Do u want to add element 1-yes, 2-no\n");
fflush(stdin);
scanf("%d",&choice1);
}while(choice1==1);


struct node *newnode2,*temp2;
int item2;
printf("LIST TWO ELEMENTS\n");
int choice2;
do
{
newnode2 =(struct node *) malloc (sizeof(struct node));
printf("Enter the data : ");
scanf("%d",&item2);
newnode2->data=item2;
newnode2->next=NULL;
if (head2==NULL)
{

head2=newnode2;

}
else
{
temp2=head2;

```

```

while(temp2->next!=NULL)
{
    temp2=temp2->next;
}
temp2->next=newnode2;
newnode2->next=NULL;

}

printf("Do u want to add element 1-yes, 2-no\n");
fflush(stdin);
scanf("%d",&choice2);
}while(choice2==1);

temp1=head1;
temp2=head2;

while(temp1->next!=NULL)
    temp1=temp1->next;

temp1->next=temp2;

printf("DISPLAY:\n");
    struct node *ptr;
ptr=head1;

if(ptr==NULL)

```

```

{
    printf("Nothing to print\n");
}
else
{
    while(ptr!=NULL)
    {
        printf("%d ",ptr->data);
        ptr=ptr->next;
    }
}
}

void Sort()
{
    struct node *newnode3,*temp3;

    int item;

    int choice;

    do
    {
        newnode3 =(struct node *) malloc (sizeof(struct node));
        printf("Enter the data : ");
        scanf("%d",&item);
        newnode3->data=item;
        newnode3->next=NULL;
        if (head3==NULL)
        {

```

```

    head3=newnode3;

}
else
{
temp3=head3;
    while(temp3->next!=NULL)
    {
        temp3=temp3->next;
    }
temp3->next=newnode3;
newnode3->next=NULL;

}

printf("Do u want to add element 1-yes,2-no\n");
fflush(stdin);
scanf("%d",&choice);
}while(choice==1);
struct node *count;
temp3=head3;
struct node *min;

int i;
while(temp3!=NULL)
{

```



```

        min=temp3;
        count=temp3;
        while(count!=NULL)
        {
            if(count->data<=min->data)
                min=count;
            count=count->next;
        }
        i=temp3->data;
        temp3->data=min->data;
        min->data=i;

        temp3=temp3->next;
    }

    printf("DISPLAY:\n");
    struct node *ptr;
    ptr=head3;

    if(ptr==NULL)
    {
        printf("Nothing to print\n");
    }
    else
    {
        while(ptr!=NULL)

```

```
{  
    printf("%d ",ptr->data);  
    ptr=ptr->next;  
}  
}  
}
```

```
int main()
```

```
{
```

```
    int choice;
```

```
    do
```

```
    {
```

```
        printf("\n1. Reverse\n2. Sorting\n3. Concatenation\n4. Exit\n");
```

```
        printf("enter choice\n");
```

```
        scanf("%d",&choice);
```

```
        switch(choice)
```

```
        {
```

```
            case 1:
```

```
                Reverse();
```

```
                break;
```

case 2:

Sort();

break;

case 3:

Concat();

case 4:

break;

default:

printf("WRong\n");

break;

}

}while(choice!=4);

return 0;

}

```
1. Reverse
2. Sorting
3. Concatenation
4. Exit
enter choice
1
Enter the data : 2
Do u want to add element 1-yes, 2-no
1
Enter the data : 2
Do u want to add element 1-yes, 2-no
1
Enter the data : 3
Do u want to add element 1-yes, 2-no
1
Enter the data : 4
Do u want to add element 1-yes, 2-no
1
Enter the data : 5
Do u want to add element 1-yes, 2-no
2
DISPLAY:
5 4 3 2 2
```

```
1. Reverse
2. Sorting
3. Concatenation
4. Exit
enter choice
2
Enter the data : 1
Do u want to add element 1-yes,2-no
1
Enter the data : 2
Do u want to add element 1-yes,2-no
1
Enter the data : 3
Do u want to add element 1-yes,2-no
1
Enter the data : 4
Do u want to add element 1-yes,2-no
1
Enter the data : 5
Do u want to add element 1-yes,2-no
2
DISPLAY:
1 2 3 4 5
```

```
1. Reverse
2. Sorting
3. Concatenation
4. Exit
enter choice
3
LIST ONE ELEMENTS
Enter the data : 1
Do u want to add element 1-yes, 2-no
1
Enter the data : 2
Do u want to add element 1-yes, 2-no
1
Enter the data : 3
Do u want to add element 1-yes, 2-no
1
Enter the data : 4
Do u want to add element 1-yes, 2-no
1
Enter the data : 5
Do u want to add element 1-yes, 2-no
2
```

```
LIST TWO ELEMENTS
Enter the data : 6
Do u want to add element 1-yes, 2-no
1
Enter the data : 7
Do u want to add element 1-yes, 2-no
1
Enter the data : 8
Do u want to add element 1-yes, 2-no
1
Enter the data : 9
Do u want to add element 1-yes, 2-no
2
DISPLAY:
1 2 3 4 5 6 7 8 9
```