1BM19CS044 merge

```
void merge( struct node *temp1,
                        struct node *temp2)
{
  struct node *newnode, *current;
     //temp = head1;
     //temp = head2;

while( temp1 != NULL &&( temp2! = NULL)
{
newnode = (struct node*) malloc((sizeof(struct node));
 newnode -> next = NULL;

 if( temp1 -> data < temp2 -> data)
  {
   newnode -> data = temp2 -> data;
    temp1 = temp1 -> next;
  }
  else
  {
   newnode -> data = temp2 -> data;
```

```
      temp1 = temp1 -> next;
    }
    if (third == NULL)
    {
        third = newnode;
        current = newnode;
    }
    else
    {
       current -> next = newnode;
       current = newnode;
    }
  }
  if (temp1 == NULL)
  {
     while (temp2 != NULL)
  newnode = (struct node*) malloc(sizeof(struct node
   newnode -> data = temp2 -> data;
     newnode -> next = NULL;
   if (third = NULL)
    {
       third = newnode;
        current = newnode;
  }
```

```
else
{
 curent -> next=newnode;
  curent=newnode;
 }
  temp2 = temp2 -> next;
 }
}
else
{
 While(temp1 != NULL)
 {
newnode=(struct node*) malloc (sizeof(struct node));
 newnode -> data = temp1 -> data;
  newnode -> next =NULL;
  if (thrd = NULL)
  {
    thrd = newnode;
    curent = newnode;
 }
 else
  {
    curent -> next= newnode;
      curent = new node;
 } temp1 = temp -> next;
  }
```

```
void create(Struct node **hptr)
{
  Struct node *newnode, *temp;
   int    item;
  Char    choice;

  do
  {
newnode = (Struct node*) malloc(sizeof(Struct node));
 Printf("Enter the data : ");
  Scanf("%d", &item);
 newnode->data = item;
 newnode->next = NULL;
Printf("Do u want to add elements \n");
 fflush(Stdin)
 Scanf("%c", &choice);
  ~~if~~
  if (*hptr == NULL) {
    *hptr = newnode;  }
    else {
   temp = *hptr;
      while(temp->next != NULL)
      {
        temp = temp->next;
      }
```

```
. Void reverse( )
{
   Struct node *prev = NULL, *current = he
                *next = NULL;
  While(current != NULL)
   {
     next = current -> next;
      current -> next = prev;
       prev = current;
      current = next;
    }
     head = prev;
 }
```