Medha

## Lab Program 5

```
struct node{
    int data;
    struct node *next;
};
void insertfirst (struct node **headptr) {
   struct node *newnode;
   int value;
   printf ("Enter value: ");
   scanf ("%d", &value);
   newnode =(struct node*) malloc (sizeof(struct node));
   ~~int value;~~
   ~~printf ("E~~
   newnode→ data = value;
   newnode→ next = NULL;
   if (*headptr == NULL)
         (*headptr) = newnode;
   else {
         newnode→next = (*headptr);
         (*headptr) = newnode;
   }
}
void insertpos (struct node ** headptr) {
   struct node *newnode, *temp;
   int count = 0, currpos = 1, value, pos;
   printf ("Enter value:");
   scanf ("%d", &value);
   newnode = (struct node*) malloc (sizeof (struct node));
   newnode → data = value;
   newnode → next = NULL;
   if (* headptr == NULL){
         (*headptr) = newnode;
   }
else {
    temp = (*headptr);
    while (temp != NULL) {
        count++;
        temp = temp->next;
    }
    printf ("There are %d elements in the list. Enter position:", count);
    scanf ("%d", &pos);
    if (pos > (count+1)) {
        printf ("No such position\n");
        return;
    }
    if (pos == count+1) {
        temp = (*headptr);
        while (temp->next != NULL)
         temp = temp->next;
        temp->next = newnode;
    }
    else {
        temp = (*headptr);
        while (temp->next != NULL) {
            if (currpos == pos-1) {
                newnode->next = temp->next;
                temp->next = newnode;
                break;
            }
            currpos++;
            temp = temp->next;
        }
    }
  }
}
```

```
void insertlast (struct node ** headptr) {
    struct node *newnode, *temp;
    int value;
    printf ("Enter value");
    scanf (" %d", &value );
    newnode = (struct node *) malloc (sizeof (struct node));
    newnode -> data = value;
    newnode -> next = NULL;
    temp = (*headptr);
    if (*headptr == NULL)
        (*headptr) = newnode;
    else {
        while (temp-> next != NUL)
            temp = temp-> next;
        temp-> next = newnode;
    }
}
void deletelast (struct node ** headptr) {
    struct node *temp;
    temp = (* headptr);
    if (*headptr == NULL)
        printf ("The list is empty\n");
    else if ((*headptr)-> next == NULL)
        (*headptr) = NULL;
    else {
        temp = *headptr;
        while ((temp-> next) -> next != NULL)
            temp = temp-> next;
        temp-> next = NULL;
    }
}
```