

## Lab-7

```
struct node
```

```
{
```

```
    int data;
```

```
    struct node* next;
```

```
};
```

```
void create(struct node** headptr, int item)
```

```
{
```

```
    struct node* newnode;
```

```
    struct node* temp;
```

```
    newnode = (struct node*) malloc(sizeof(struct node));
```

```
    newnode->data = item;
```

```
    newnode->next = NULL;
```

```
    temp = *headptr;
```

```
    if(temp == NULL)
```

```
        *headptr = newnode;
```

```
    else
```

```
    {
```

```
        while(temp->next != NULL)
```

```
        {
```

```
            temp = temp->next;
```

```
        }
```

```
    }
```

```
}
```

```
void concat(struct node* temp1, struct node* temp2)
```

```
{
```

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

```
        temp1 = temp1->next;
```

```
    temp1->next = temp2;
```

```
}
```

```
void sort (struct node **h)
{
```

```
    int a;
```

```
    struct node * temp1;
```

```
    struct node * temp2;
```

```
    for (temp1 = *h, temp1 != NULL; temp1 = temp1->next;
```

```
        for (temp2 = temp1->next; temp2 != NULL;
```

```
            temp2 = temp2->next)
```

```
    {
```

```
        if (temp2->data < temp1->data)
```

```
        {
```

```
            a = temp1->data;
```

```
            temp1->data = temp2->data;
```

```
            temp2->data = a;
```

```
        }
```

```
    }
```

```
}
```

```
void reverse (struct node **head)
{
```

```
    struct node * prev = NULL, * current = *head,
```

```
    * next = NULL;
```

```
    while (current != NULL)
```

```
    {
```

```
        next = current->next;
```

```
        current->next = prev;
```

```
        prev = current;
```

```
        current = next;
```

```
    }
```

```
    *head = prev;
```

```
}
```