Name : S Skanda
USN : 1BM19CS137

Date : 7/12/2020

```
#include <stdio.h>
#include <stdlib.h>
#include <process.h>
struct node
{ int info;
  struct node *link;
};
typedef struct node *NODE;
NODE getnode()
{ NODE x;
  x = (NODE) malloc (sizeof (struct node));
  if (x == NULL)
  { printf("Memory full\n");
    exit(0);
  }
  return x;
}
NODE insert rear (NODE first, int item)
{ NODE temp, cur;
  temp = getnode();
  temp->info = item;
  temp->link = NULL
  if (first == NULL)
     return temp;
  cur = first;
  while (cur->link != NULL)
    { cur=cur->link;
      cur->link = temp;
      return first;
    }
```

```
Node delete_front (NODE first)
{ NODE temp;
  if (first == NULL)
  { printf ("List is empty cannot delete \n");
    return first;
  }
  temp = first;
  temp = temp -> link;
  printf ("Item deleted at front is %d"
                                  first -> info);
  free (first)
  return temp;
}
void display (NODE first)
{ NODE temp;
  if (first == NULL)
     printf ("List is empty");
  for (temp = first; temp != NULL; temp = temp -> link)
  { printf ("%d \n" temp -> info);
}
NODE concat (NODE first, NODE second)
{ NODE cur;
  if (first == NULL)
       return second;
  if (second == NULL)
       return first;
  cur = first
  while (cur -> link != NULL)
     cur = cur -> link;
  cur -> link = second;
  return first;
}
```

```
NODE reverse (NODE first)
{ NODE cur, temp;
  cur = NULL;
  while (first != NULL)
  { temp = first;
    first = first -> link;
    temp -> link = cur;
    cur = temp;
  }
  return cur;
}
NODE sortList (NODE first){
 NODE current = first, index = NULL;
 int temp;
 if (first == NULL)
 { printf ("List is empty");
   return current;
 }
 else {
    while (current != NULL) {
    index = current -> link;
    while (index != NULL) {
       { if ( current -> info > index -> info)
          { temp = current -> info;
            current -> info = index -> info;
            index -> info = temp;
          }
       index = index -> link;
       }
    current = current -> link;
    }
   return current
 }
}
```

```
int main()
{ int item, choice, pos, i, n;
  NODE first = NULL, a, b;
  for(;;)
  { printf("1. insertfront 2: concat 3: reverse 4: order
           5: display 6: delete 7: exit\n");
    printf("Enter the choice:");
    scanf("%d", &choice);
    switch(choice)
    { switch:
      case 1: printf("Enter the item:");
              scanf("%d", item:");
              first = insert_rear(first, item);
              break;
      case 2: printf("Enter no of nodes in list:");
              scanf("%d", &n);
              a = NULL;
              for(i=0; i<n; i++)
              { printf("Enter an item:");
                scanf("%d", &item);
                a = insert_rear(a, item);
              }
              first = concat(first, a);
              display(first);
              break;
      case 3: first = reverse(first)
              display(first);
              break;
      case 4: sortlist(first);
              display(first);
              break
      case 5: display(first); break;
      default: exit(0);
  }
}
```