**DS/linkedlist.c**

```c
//implement linked list data structure in c
#include <stdio.h>
#include <stdlib.h>

typedef struct linkedlist {
   int data;
   struct linkedlist *next;
}lkdlist;

int isEmpty(lkdlist *head){
   if(!head){
      return 1;
   }
   return 0;
}

int len(lkdlist *head){
   int count = 0;
   while(head){
      head = head->next;
      count++;
   }
   return count;
}

void push(lkdlist **head, int new_data, int index){
   if(len(*head) < index || index < 0){
      printf("Index Out of Range %d\n",len(*head));
      return;
   }

   lkdlist *new_node = (lkdlist*)malloc(sizeof(lkdlist));
   new_node->data = new_data;

   if(index == 0){
      new_node->next = *head;
      *head = new_node;
      return;
   }
   int i = 0;
   lkdlist *temp = *head, *prev;
   do{
      prev = temp;
      temp = temp->next;
      i++;
   }while(i < index);
   prev->next = new_node;
   new_node->next = temp;
}

int pop_at(lkdlist **head, int index){
   if(isEmpty(*head)){
      printf("No Element Found.\n");
      return -1;
   }

   if(len(*head) < index || index < 0){
      printf("Index Out of Range.\n");
      return -1;
   }
```

```
    lkdlist *temp = *head, *prev;
    int i = 0;
    int data;
    if(i == index){
        data = temp->data;
        *head = temp->next;
        free(temp);
        return data;
    }

    while(i < index){
        prev = temp;
        temp = temp->next;
        i++;
    }
    prev->next = temp->next;
    data = temp->data;
    free(temp);
    return data;
}

void display(lkdlist *head) {
    while (head != NULL) {
        printf("%d -> ", head->data);
        head = head->next;
    }
    printf("NULL\n");
}

int main(){
    lkdlist *st1 = NULL, *st2 = NULL;
    int n = 20;
    for(int i = 1; i < n; i+=2){
        push(&st1,i,0);
    }
    for(int i = 0; i < n; i+=2){
        push(&st2,i,0);
    }
    display(st1);
    display(st2);
    return 0;
}
```

## OUTPUT

```
PS S:\WorkSpace\CollegeWork\DataStructure> gcc .\linkedlist.c
PS S:\WorkSpace\CollegeWork\DataStructure> ./a
19 -> 17 -> 15 -> 13 -> 11 -> 9 -> 7 -> 5 -> 3 -> 1 -> NULL
18 -> 16 -> 14 -> 12 -> 10 -> 8 -> 6 -> 4 -> 2 -> 0 -> NULL
PS S:\WorkSpace\CollegeWork\DataStructure>
```