DS/linkedlist.c

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

typedef struct circular_linked_list{
    int data;
    struct circular_linked_list *next;
}cl_list;

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

int len(cl_list *head){
    int count = 0;
    if(!head){
        return 0;
    }
    cl_list *temp = head;
    while(temp ->next != head){
        temp = temp->next;
        count++;
    }
    return count+1;
}

void push(cl_list **head, int new_data, int index){
    cl_list *new_node = (cl_list*)malloc(sizeof(cl_list));
    new_node->data = new_data;

    if(index > len(*head) || index < 0){
        printf("idnex out of range.\n");
        return;
    }
    if(!*head){
        new_node->next = *head;
        *head = new_node;
    }
    cl_list *temp = *head, *prev;

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

int pop_at(cl_list **head, int index){
    if(len(*head) <= index || !*head || index < 0){
        printf("index out of range or list is empty.\n");
```

```
        return -1;
    }

    cl_list *temp = *head, *prev;
    int i = 0;
    int data;
    if(i == index){
        data = temp->data;
        while(temp->next != *head){
            temp = temp->next;
        }
        prev = *head;
        *head = prev->next;
        temp->next = *head;
        free(prev);
        return data;
    }
    while(i < index){
        prev = temp;
        temp = temp->next;
        i++;
    }
    prev->next = temp->next;
    data = temp->data;
    free(temp);
    return data;
}

void display(cl_list *head){
    if(!head){
        printf("List is Empty.(No Element Found)\n");
        return;
    }
    cl_list *temp = head;
    while(temp->next != head){
        printf("%d -> ",temp->data);
        temp = temp->next;
    }
    printf("%d\n",temp->data);
}

void main(){
    cl_list *ll1 = NULL, *ll2 = NULL;
    int n = 20;
    for(int i = 0; i < n; i+=2){
        push(&ll1,I,0);
    }
    for(int i = 1; i < n; i+=2){
        push_top(&ll2,i);
    }
    display(ll1);
    display(ll2);
}
```

OUTPUT

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