DS/cdlinkedlist.c

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

typedef struct circular_doubly_linked_list{
    int data;
    struct circular_doubly_linked_list* next;
    struct circular_doubly_linked_list* prev;
}cdl_list;

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

int len(cdl_list *head){
    if(!head){
        return 0;
    }
    int count = 1;
    cdl_list *temp = head;
    while(temp->next != head){
        count++;
        temp = temp->next;
    }
    return count;
}
void push(cdl_list **head, int new_data, int index){
    int l = len(*head);
    if(l < index || index < 0){
        printf("Index out of range.\n");
        return;
    }
    cdl_list *new_node = (cdl_list*)malloc(sizeof(cdl_list));
    new_node->data = new_data;
    if(l == index || index == 0){
        if(!*head){
            new_node->next = new_node;
            new_node->prev = new_node;
            *head = new_node;
            return;
        }
        new_node->next = *head;
        (*head)->prev->next = new_node;
        new_node->prev = (*head)->prev;
        (*head)->prev = new_node;
        if(index == 0){
            *head = new_node;
        }
        return;
    }
    int i = 0;
    cdl_list *temp = *head, *prev;
    while(i < index){
        prev = temp;
        temp = temp->next;
        i++;
    }
    prev->next = new_node;
    new_node->prev = prev;
    temp->prev = new_node;
    new_node->next = temp;
}
```

```
int pop_at(cdl_list **head, int index){
    int l = len(*head);
    if(l <= index || index < 0){
        printf("Index out of range.\n");
        return -1;
    }
    int data;
    cdl_list *temp, *prev;
    if(index == 0 || index == l-1){
        if((*head)->next == *head){
            data = (*head)->data;
            *head = NULL;
            return data;
        }
        if(index == 0){
            temp = *head;
            *head = (*head)->next;
        }else{
            temp = (*head)->prev;
        }
        prev = temp->prev;
        prev->next = (*head);
        (*head)->prev = prev;
        data = temp->data;
        free(temp);
        return data;
    }
    int i = 0;
    temp = (*head);
    while(i < index){
        prev = temp;
        temp = temp->next;
        i++;
    }
    prev->next = temp->next;
    temp->next->prev = prev;
    data = temp->data;
    free(temp);
    return data;
}
void displayr(cdl_list *head){
    if(!head){
        printf("List is Empty.(No Element Found)\n");
        return;
    }
    cdl_list *temp = head->prev;
    while(temp != head){
        printf("%d <-> ",temp->data);
        temp = temp->prev;
    }
    printf("%d\n",temp->data);
}
void main(){
    cdl_list *ll1 = NULL;
    for(int i = 0; i < 20; i+=2){
        push(&ll1,i,0);
    }
    display(ll1);
}
```

OUTPUT

```
PS S:\WorkSpace\CollegeWork\DataStructure> gcc .\cdlinkedlist.c
PS S:\WorkSpace\CollegeWork\DataStructure> ./a
18 <-> 16 <-> 14 <-> 12 <-> 10 <-> 8 <-> 6 <-> 4 <-> 2 <-> 0
PS S:\WorkSpace\CollegeWork\DataStructure>
```