#include #include struct node { int data; struct node*next; int visit; }; int init(struct node**head) { *head=NULL; return 1; } int destroy(struct node**head) { *head=NULL; return 1; } int insertcll(struct node**head,int data,int predata) { struct node*newnode=(struct node*)malloc(sizeof(struct node)); if(newnode==NULL) { return 0; }newnode->data=data; struct node*ptr=*head; if(*head==NULL) { *head=newnode; newnode->next=newnode; } while(ptr!=NULL && ptr->data!=predata) { ptr=NULL; ptr=ptr->next; } if(ptr==NULL) { return 0; } else { newnode->next=ptr->next; ptr->next=newnode; return 1; } } int traverse(struct node*head) { if(head==NULL) { return 0; } struct node*ptr=head; printf("%d ",ptr->data); ptr=ptr->next; while (ptr!=NULL && ptr!=head) { printf("%d ",ptr->data); ptr=ptr->next; } printf("\n"); return 1; }int isCircular(struct node*head) { if(head==NULL) { return 0; } struct node*ptr=head; while (ptr!=NULL && ptr->visit!=1) { ptr->visit=1; ptr=ptr->next; } if(ptr==NULL) { printf("It is not circular \n"); return 0; } else { printf("It is circular \n"); return 1; } } int main() { struct node* head =(struct node*)malloc(sizeof(struct node)); init(&head); int n; int data1; int predata; printf("enter the no.of elements:"); scanf("%d",&n); // inserting as circular linear linked lists for (int i=0;i