Question 1

#include<stdio.h>

#include<stdlib.h>

struct node{

int data;

struct node \* next;

};

struct node \* head= NULL;

void create()

{

int d;

printf("enter the data: ");

scanf("%d",&d);

struct node \* n=(struct node \*) malloc( sizeof(struct node));

n->data=d;

n->next=NULL;

if(head==NULL)

{

head=n;

head->next=head;

return;

}

if(head->next==head)

{

head->next=n;

n->next=head;

return;

}

struct node\* temp=head->next;

while(temp->next!=head)

{

temp=temp->next;

}

n->next=temp->next;

temp->next=n;

}

void print()

{

if(head==NULL) {printf("NULL");return;}

printf("->%d->",head->data);

struct node \* temp = head->next;

while(temp!=head)

{

printf("%d->",temp->data);

temp=temp->next;

}

printf("\n");

}

int main()

{

int n;

printf("enter the length of circular link list : ");

scanf("%d",&n);

for(int i=0;i<n;i++)

{

create();

}

print();

}

Question 2

#include<iostream>

using namespace std;

class node{

public:

int data;

node \* next;

node(int d)

{

this->data=d;

this->next=NULL;

}

void insertion\_at\_end(node \* & head,int d)

{

node \* n = new node (d);

if(head==NULL)

{

n->next=n;

head=n;

return;

}

node\* temp=head->next;

while(temp->next!=head)

{

temp=temp->next;

}

n->next=temp->next;

temp->next=n;

}

void insertion\_at\_beg(node \* & head,int d)

{

node \* n = new node (d);

if(head==NULL)

{

n->next=n;

head=n;

return;

}

node\* temp=head->next;

while(temp->next!=head)

{

temp=temp->next;

}

n->next=temp->next;

temp->next=n;

head=n;

}

void in\_at\_pos(node \* & head, int d,int pos)

{

if(pos==1)

{

head->insertion\_at\_beg(head,d);

return;

}

node \* n= new node(d);

node\* temp=head;

int ct=1;

while(ct<pos-1)

{

ct++;

temp=temp->next;

}

n->next=temp->next;

temp->next=n;

}

void print(node \* & head)

{

node\* temp=head->next;

cout<<"->"<<head->data<<"->";

while(temp!=head)

{

cout<<temp->data<<"->";

temp=temp->next;

}

cout<<"\n";

}

};

int main()

{

node\* head=NULL;

int t=1;

while(t)

{

cout<<"enter 1 to insert at beg \nenter 2 to insert at any pos \nenter 3 to insert at end\nenter 4 to print \nenter 0 to exit\n";

cin>>t;

int n,pos;

switch(t)

{

case 1:

cout<<"enter data : ";

cin>>n;

head->insertion\_at\_beg(head, n);

break;

case 2:

cout<<"enter data : ";

cin>>n;

cout<<"enter pos : ";

cin>>pos;

head->in\_at\_pos(head,n,pos);

break;

case 3:

cout<<"enter data : ";

cin>>n;

head->insertion\_at\_end(head, n);

break;

case 4:

head->print(head);

break;

}

}

}

Question 3

#include<stdio.h>

#include<stdlib.h>

struct node{

int data;

struct node \* next;

};

struct node \* head= NULL;

void create()

{

int d;

printf("enter the data: ");

scanf("%d",&d);

struct node \* n=(struct node \*) malloc( sizeof(struct node));

n->data=d;

n->next=NULL;

if(head==NULL)

{

head=n;

head->next=head;

return;

}

if(head->next==head)

{

head->next=n;

n->next=head;

return;

}

struct node\* temp=head->next;

while(temp->next!=head)

{

temp=temp->next;

}

n->next=temp->next;

temp->next=n;

}

void search()

{

if(head==NULL) {printf("under flow\n");return;}

int d;

printf("enter the number to search : ");

scanf("%d",&d);

if(head->data==d){printf("the element is present\n");return ;}

struct node \* temp = head->next;

while(temp!=head)

{

if(temp->data==d){

printf("the element is present\n");

return;

}

temp=temp->next;

}

printf("not present\n");

}

void print()

{

if(head==NULL) {printf("NULL");return;}

printf("->%d->",head->data);

struct node \* temp = head->next;

while(temp!=head)

{

printf("%d->",temp->data);

temp=temp->next;

}

printf("\n");

}

int main()

{

int n;

printf("enter the length of circular link list : ");

scanf("%d",&n);

for(int i=0;i<n;i++)

{

create();

}

search();

}

Question 4

#include<stdio.h>

#include<stdlib.h>

struct node{

int data;

struct node \* next;

};

struct node \* head= NULL;

void create()

{

int d;

printf("enter the data: ");

scanf("%d",&d);

struct node \* n=(struct node \*) malloc( sizeof(struct node));

n->data=d;

n->next=NULL;

if(head==NULL)

{

head=n;

return;

}

struct node\* temp=head;

while(temp->next!=NULL)

{

temp=temp->next;

}

temp->next=n;

}

void sort()

{

struct node\* i=head;

while(i!=NULL)

{

struct node\*j=head;

while(j->next!=NULL)

{

if(j->data > j->next->data)

{

int temp=j->data;

j->data=j->next->data;

j->next->data=temp;

}

j=j->next;

}

i=i->next;

}

}

void print()

{

if(head==NULL) {printf("NULL");return;}

struct node \* temp = head;

while(temp!=NULL)

{

printf("%d->",temp->data);

temp=temp->next;

}

printf("NULL\n");

}

int main()

{

int n;

printf("enter the length of circular link list : ");

scanf("%d",&n);

for(int i=0;i<n;i++)

{

create();

}

print();

sort();

print();

}