

توضیحات سوال ۱ پروژه ۴:

علیرضا سعیدنیا شماره دانشجویی ۴۰۰۱۰۸۳۳

کلاس ند را تعریف میکنیم که شامل فیلدهای دیتا و ند بعدی است

در سوال اشاره نشده یک طرفه یا دو طرفه پس یک طرفه در نظر گرفتیم

یک کانستراکتور تعریف میکنیم که وقتی داده را وارد کردیم به عنوان فیلد داده ای وارد ند شود

```
public class Node{
    int data;
    Node next;
    public Node(int data) {
        this.data = data;
    }

    public Node() {
    }
}
```

گره راسی را تعریف میکنیم و آن را هد مینامیم ، چون در جاوا اینتیجر نمیتواند نال باشد بنابراین به آن مقدار ۰ را میدهیم که نشانه نال است ([منبع استک اور فلو](#))

```
public Node head=new Node( data: 0) ;
public Node tail = head;
```

هد را هم مساوی تیل میذاریم چون اگر لیست خالی باشد و کاربر نخواهد عددی وارد کند یک گره راسی به مقدار صفر داریم و در خط ۷۸ برنامه مینویسیم لیست خالی است ولی گره راسی موجود است (لیست ذاتا خالی است ولی گره راسی دارد) ، اگر خالی نباشد از کاربر میپرسیم چندتا عدد میخواهد وارد کند به تعداد آن عدد عدد وارد میکنیم و در متد اد آنها را عدد میکنیم

```
public void add(int data){
    Node newNode = new Node(data);
    //tail will point to new node.
    tail.next = newNode;
    //New node will become new tail.
    tail = newNode;
    tail.next = head;
}
```

در متد اد میاید و ند جدید با مقدار داده ای که دادیم را میسازد ، تیل میاید و به ند جدید اشاره میکند و ند جدید همان تیل میشود یعنی اخر لیست میشود ، چون لیست حلقوی است ند اخر لیست خانه بعدی اش به ند ابتدایی و هد اشاره میکند

```
public static int findMin(Node head){
    int a=head.next.data;
    Node cur=head.next;
    while(cur!=head){
        if(a<cur.data){
            cur=cur.next;
        }
        else{
            a=cur.data;
            cur=cur.next;
        }
    }
    return a;
}
```

برای پیدا کردن کمترین عدد در واقع باید لیست را پیمایش کنیم ، پیمایش لیست را طبق اسلاید استاد رفتم و یک متغیر کر تعریف کردم که ان تا انتهای لیست میروود و تا زمانی که برابر با هد یعنی گره راسی نشده است میاید و پیمایش و مقایسه را برای پیدا کردن کوچکترین عدد انجام میدهد و اولین عدد را به عنوان مقایسه کننده میگذارد و اگر عددها کوچکتر بود میاید داخل آ قرار میدهد.

```
public static void searchAddress(int a,Node head){
    Node cur=head.next;
    while(cur!=head){
        if(a==cur.data){
            System.out.println("address of the min value is "+cur);
            return;
        }
        else{
            cur=cur.next;
        }
    }
}
```

پس از پیدا کردن کوچکترین عدد در لیست پیوندی و انجام دادن پیمایش حال نوبت به نمایش آدرس آن میرسد، در نمایش آدرس باید بیایم جستجو کنیم و اگر مقدار یک ند برابر با مقدار عدد ما بود آن ند را آدرسش را نمایش دهیم

در آخر نوبت به نمایش لیست پیوندی و در صورت نیاز ثابت کردن اینکه این لیست یک لیست حلقوی است میرسد

```

public void display() {
    Node current = head;
    System.out.println("next data after tail is " +tail.next.data);

    System.out.println("Nodes of the circular linked list: ");
    do{
        System.out.print(" " + current.data);
        current = current.next;
    }while(current != head);
    System.out.println();
}

```

در متد مین هم میاییم و عدد هارا میگیریم اگر مشکلی بود کچ میشود و عددهارا اضافه به لیستمان میکنیم و نمایش میدهیم و جستجو میکنیم

```

public static void main(String[] args) {
    try {
        Scanner myScanner=new Scanner(System.in);
        main cl = new main();
        System.out.println("how many numbers do you want to enter?");
        int many=myScanner.nextInt();
        if(many==0){
            System.out.println("list is empty but there is gere rasi");
            System.exit( status: 0);
        }
        for(int i=0;i<many;i++){
            cl.add(myScanner.nextInt());
        }
        System.out.println("min value is "+findMin(cl.head));
        searchAddress(findMin(cl.head), cl.head);
        cl.display();
    }
    catch (Exception e){

    }
}

```

نمونه خروجی :

min value is 2

address of the min value is main\$Node@7530d0a

next data after tail is 0

Nodes of the circular linked list:

0 421 44 23542 3423 2