

Max Min

Problem Statement

You need to take a singly linked list of integer values as input and print the maximum and minimum value of the singly linked list.

Note: You must use a singly linked list, otherwise you will not get marks.

Input Format

- Input will contain the values of the singly linked list, and will terminate with -1.

Constraints

1. $1 \leq N \leq 1000$; Here N is the maximum number of nodes of the linked list.
2. $0 \leq V \leq 1000$; Here V is the value of each node.

Output Format

- Output the maximum value then the minimum value.

Sample Input 0

2 4 1 5 3 6 -1

Sample Output 0

6 1

Sample Input 1

2 -1

Sample Output 1

2 2

Same to Same

Problem Statement

You will be given two singly linked list of integer values as input. You need to check if all the elements of both list are same which means both list are same. If they are same print "YES" otherwise print "NO".

Note: You must use singly linked list, otherwise you will not get marks.

Input Format

- First line will contain the values of the first singly linked list, and will terminate with -1.
- Second line will contain the values of the second singly linked list, and will terminate with -1.

Constraints

1. $1 \leq N1, N2 \leq 1000$; Here $N1$ and $N2$ is the maximum number of nodes of the first and second linked list.
2. $0 \leq V \leq 1000$; Here V is the value of each node.

Output Format

- Output "YES" or "NO".

Sample Input 0

10 20 30 40 -1

10 20 30 40 -1

Sample Output 0

YES

Sample Input 1

10 20 30 40 -1

10 20 30 -1

Sample Output 1

NO

Sample Input 2

10 20 30 40 -1

40 30 20 10 -1

Sample Output 2

NO

Get Me Mid

Problem Statement

You need to take a singly linked list of integer value as input. Then you need to sort the linked list in descending order and print the middle element of it. If there are multiple values in the middle, print both.

Note: You must use singly linked list, otherwise you will not get marks.

Input Format

- Input will contain the values of the singly linked list, and will terminate with -1.

Constraints

1. $1 \leq N \leq 1000$; Here N is the maximum number of nodes of the linked list.
2. $0 \leq V \leq 1000$; Here V is the value of each node.

Output Format

- Output the value/values of the middle element.

Sample Input 0

```
20 10 40 30 50 -1
```

Sample Output 0

Sample Input 1

20 30 10 40 60 50 -1

Sample Output 1

40 30

Sample Input 2

726 -1

Sample Output 2

726

Sample Input 3

348 726 -1

Sample Output 3

726 348

Reverse and Original

Problem Statement

You need to take a singly linked list of integer value as input. Then you need to print the singly linked list in reverse order, after that print the original linked list also.

Note: You must use singly linked list, otherwise you will not get marks.

Input Format

- Input will contain the values of the singly linked list, and will terminate with -1.

Constraints

1. $1 \leq N \leq 10^6$; Here N is the maximum number of nodes of the linked list.
2. $0 \leq V \leq 10^9$; Here V is the value of each node.

Output Format

- In first line, output the linked list in reverse order.
- In second line, print the original linked list.

Sample Input 0

```
10 20 30 40 50 -1
```

Sample Output 0

```
50 40 30 20 10  
10 20 30 40 50
```

Sample Input 1

```
20 40 50 10 -1
```

Sample Output 1

10 50 40 20

20 40 50 10

Queries

Problem Statement

You have a singly linked list which is empty initially. Then you will be given Q queries. In each query you will be given two values X and V .

- If X is 0 that means you will insert the value V to the head of the linked list.
- And if X is 1 then you will insert the value V to the tail of the linked list.
- After that for each query you need to print the value of the head and tail of the list.

Note: You must use singly linked list, otherwise you will not get marks.

Input Format

- First line will contain Q .
- Next Q lines will contain X and V .

Constraints

1. $1 \leq Q \leq 10^6$;
2. $0 \leq X \leq 1$;
3. $0 \leq V \leq 10^9$

Output Format

- For each query output the value of head and tail of the linked list.

Sample Input 0

5

0 10

0 20

1 30

0 40

1 50

Sample Output 0

10 10

20 10

20 30

40 30

40 50

Explanation 0

Query 1: 10 is inserted at head. Linked list looks like: 10->NULL. Now both head and tail are 10.

Query 2: 20 is inserted at head. Linked list looks like: 20->10->NULL. Now head is at 20 and tail is at 10.

Query 3: 30 is inserted at tail. Linked list looks like: 20->10->30->NULL. Now head is at 20 and tail is at 30.

Query 4: 40 is inserted at head. Linked list looks like: 40->20->10->30->NULL. Now head is at 40 and tail is at 30.

Query 5: 50 is inserted at tail. Linked list looks like: 40->20->10->30->50->NULL. Now head is at 40 and tail is at 50.

Sample Input 1

3

1 10

1 20

0 30

Sample Output 1

10 10

10 20

30 20

Explanation 1

Query 1: 10 is inserted at tail. Linked list looks like: 10->NULL. Now both head and tail are 10.

Query 2: 20 is inserted at tail. Linked list looks like: 10->20->NULL. Now head is at 10 and tail is at 20.

Query 3: 30 is inserted at head. Linked list looks like: 30->10->20->NULL. Now head is at 30 and tail is at 20.

