Instructions:

1. Please write the program in C language.
2. Use the program name as "$A\langle assignmentNo\rangle\_\langle rollNo\rangle$.c". For example A3_15XXXXXX.c for the program of assignment-3 submitted by 15XXXXXX. If a group is submitting the assignment, then the program name should be "$A\langle assignmentNo\rangle\_\langle rollNo1\rangle\_\langle rollNo2\rangle$.c".
3. Upload the latest program in your google drive and share it to "**assignments.ds@gmail.com**". The procedure is as follows.
   - Right click the uploaded file and click "share".
   - Select "Advanced" button at the bottom right corner.
   - Give "assignments.ds@gmail.com" as "view access".
   - Untick the Checkbox for "Notify People" in order to avoid email notification.
   - Click "Ok" and then "Done".
4. The code can be deleted and submitted again with the same name, any time before 11:59 pm of the last date of submission. The latest file will be used for evaluation. Any file submitted after the deadline will not be evaluated.
5. As the file is only shared as view access to the evaluator, students are not supposed to delete the file in their own drive after the deadline, which may cause deletion of the shared file in evaluator's drive. Hence keep the files in the drive unchanged till the end of the semester. This is required to maintain the ownership of the file.
6. An Input test case file is also attached to the assignment notification email. This input can be given to the program using the following syntax.
   ./a.out < $\langle inputFileName\rangle.txt$
   The code snippet for reading this input is given below.
7. Please refrain from copying the code.
8. Last date of submission: **19th September 2016 11:59 pm.**

Code Snippet to read the input test case

```
#include<stdio.h>

void main(){
    int n,i;
    char c[20][100];
    scanf("%d\n",&n);

    printf("\nThe number of lines in the input file, n = %d",n);

    for(i=0;i<n;i++){
        gets(c[i]);
        printf("\nc[%d]=%s",i,c[i]);
```

```
    }
}
```

Question

Implement **online insertion sort algorithm** using **doubly linked list** as per the following criteria.

- The number of subsequent input lines is given in the first line of input file.
- Each line in the input file can contain either an integer or the alphabet "P".
- Each element(integer) should be stored in a separate node and the node should be placed in the appropriate position in the doubly linked list as per the insertion sort algorithm.
- The moment you see the alphabet "P", print the sorted list of elements inserted so far (Assume that you would be asked to print the sorted list only after at least one number is added to the linked list). Once the input is exhausted, print the final list of sorted elements.

Sample Input

```
12
7
1
P
2
4
P
5
9
25
P
11
10
```

Output

```
1 7
1 2 4 7
1 2 4 5 7 9 25
Final sorted list : 1 2 4 5 7 9 10 11 25
```

The End.