Consider a list (`list = []`). You can perform the following commands:

1. `insert i e`: Insert integer e at position i.
2. `print`: Print the list.
3. `remove e`: Delete the first occurrence of integer e.
4. `append e`: Insert integer e at the end of the list.
5. `sort`: Sort the list.
6. `pop`: Pop the last element from the list.
7. `reverse`: Reverse the list.

Initialize your list and read in the value of n followed by n lines of commands where each command will be of the 7 types listed above. Iterate through each command in order and perform the corresponding operation on your list.

Example

$N = 4$

append 1

append 2

insert 3 1

print

- **append 1**: Append 1 to the list, $arr = [1]$.
- **append 2**: Append 2 to the list, $arr = [1, 2]$.
- **insert 3 1**: Insert 3 at index 1, $arr = [1, 3, 2]$.
- **print**: Print the array.

  Output:

```
[1, 3, 2]
```

Input Format

The first line contains an integer, n, denoting the number of commands. Each line i of the n subsequent lines contains one of the commands described above.

Constraints

- The elements added to the list must be integers.

Output Format

For each command of type `print`, print the list on a new line.

Sample Input 0

```
12
insert 0 5
insert 1 10
insert 0 6
print
remove 6
append 9
append 1
sort
print
pop
reverse
print
```

Sample Output 0

```
[6, 5, 10]
[1, 5, 9, 10]
[9, 5, 1]
```

Change Theme Language Python 3

```
if __name__ == '__main__':
    n = int(input())
    a=[]
    for i in range(n):
        x=input().split()
        if x[0]=="insert":
            a.insert(int(x[1]),int(x[2]))
        elif x[0]=="print":
            print(a)
        elif x[0]=="remove":
            a.remove(int(x[1]))
        elif x[0]=="append":
            a.append(int(x[1]))
        elif x[0]=="sort":
            a.sort()
        elif x[0]=="pop":
            a.pop()
        elif x[0]=="reverse":
            a.reverse()

```

Line: 20 Col: 1

Upload Code as File Test against custom input Run Code Submit Code

Congratulations

You solved this challenge. Would you like to challenge your friends?

Next Challenge

Test case 0

Test case 1

Compiler Message

Success

Input (stdin) Download

```
12
insert 0 5
insert 1 10
insert 0 6
print
remove 6
append 9
append 1
sort
```