HOMEWORK 5

Write a C program to implement the following requirement:

Input:

The program will read from standard input any text up to 10,000 characters and store each word (a string that does not contain any whitespace with a maximum of 100 characters) into a tree node of a Binary Search Tree (BST), using the following struct:

```
struct TREENODE {
    char *word;
    Struct TREENODE *parent
    struct TREENODE *left;
    struct TREENODE *right;
};
```

The word should be converted into **lowercase** before adding to the BST.

In this BST, each node stores a **unique** word (duplicated words are not added to the tree).

Output:

The program will print to standard output 2 things:

- The list of words in **reverse alphabetical** order, each word is separated by a single comma ",".
- The list of words after removing words that start with a vowel (a, e, i, o, u) in **alphabetical** order, each word is separated by a single comma ",".

Note: If there is no word in the input text, the program must print empty string to stdout.

REQUIREMENT:
You must implement **addNode**, and **deleteNode** functions for the BST.
Otherwise, you will get **0**.

SAMPLE INPUT 1
```
THIS is a huge elephant.
```

SAMPLE OUTPUT 1
```
this,is,huge,elephant,a
huge,this
```

SAMPLE INPUT 2
```
First line is      here


Second line is here
           Third line is here
```

SAMPLE OUTPUT 2
```
third,second,line,is,here,first
first,here,line,second,third
```