Solution A

Problem Statement
The goal is to read in a list of students and exam scores into an array and output the name of the student with the median score, that is, the one which if you put them all in order would be in the middle. For example the median of 40, 80, 20, 30 and 50 is 40, because if you put them all in order 40 would be the middle value, so output the name of the student who got 40.
You can assume that there is always an odd number of students, and they all have different scores.

Input Format
The first line contains *N*, the number of inputs. This is followed by *N* pairs of names and integers separated by a space, each on a separate line.
Output Format
The name of the student who represents the median.
Constraints
$1 \leq N \leq 1000$
$0 \leq n \leq 100$
Sample Input
5
Eoin 18
Claire 94
David 34
Dylan 69
John 25
Sample Output
David

```
import java.util.*;

public class Solution {
    public static void main(String args[] ) throws Exception {
        Scanner myscanner = new Scanner(System.in);
        int num = Integer.parseInt(myscanner.nextLine());
        String[] array = new String[num];
        for(int i=0;i<num;i++){
            array[i]=myscanner.nextLine();
        }
        for(int i=num-1;i>0;i--){
            for(int j=0;j<i;j++){
                if(getNum(array[j])>getNum(array[j+1])){
                    String temp=array[j];
                    array[j]=array[j+1];
                    array[j+1]=temp;
                }
            }
        }
        System.out.println(getString(array[(num-1)/2]));
    }

    public static int getNum(String input){
        for(int i=0;i<input.length();i++){
            if(input.charAt(i)==' '){
                return Integer.parseInt(input.substring(i+1));
            }
        }
        return -1;
    }

    public static String getString(String input){
        for(int i=0;i<input.length();i++){
            if(input.charAt(i)==' '){
                return input.substring(0,i);
            }
        }
        return "";
    }
}
```

Solution B

Problem Statement
The goal is to sort a list of words firstly by the length of the word, with shorter words coming first. For words that have the same length (e.g. all words of length 5), sort them by alphabetical order.

Input Format
The first line contains N (number of words) followed by N lines. Each line contains a String.

Output Format
A line consisting of the the words sorted in alphabetical order, each separated by a space.

Constraints
$1 \leq N \leq 100$

Sample Input
4
one
two
three
four

Sample Output
one two four three

```
import java.util.*;

public class Solution {
    public static void main(String args[] ) throws Exception {
        Scanner myscanner = new Scanner(System.in);
        int num = Integer.parseInt(myscanner.nextLine());
        String[] array = new String[num];
        for(int i=0;i<num;i++){
            array[i]=myscanner.nextLine();
        }
        for(int i=num-1;i>0;i--){
            for(int j=0;j<i;j++){
                if(check(array[j],array[j+1])){
                    String temp=array[j];
                    array[j]=array[j+1];
                    array[j+1]=temp;
                }
            }
        }
        for(int i=0;i<num;i++){
            System.out.print(array[i]);
            System.out.print(" ");
        }
    }

    public static boolean check(String one, String two){
        if(one.length()>two.length()){
            return true;
        }else if(one.length()<two.length()){
            return false;
        }else if(one.compareTo(two)>0){
            return true;
        }else{
            return false;
        }
    }
}
```