

# Sort Character

## Problem

Sort Character

## Input

One line of words (S)

## Output

Contains vowel and consonant characters that has been sorted according to the following rules. - Sort the letters according to the order they came out - Separate between vowels and consonants. - Process as lowercase letters (whitespaces are ignored)

## Test Case

Input	Output
Input one line of words (S) : Sample Case	Vowel Characters : aaee Consonant Characters : ssmplc
Input one line of words (S) : Next Case	Vowel Characters : aee Consonant Characters : nxtcs

## Source Code

```
import java.util.Arrays;

public class SortCharacter {
    public static void main(String[] args) {
        String input1 = "Sample Case";
        String input2 = "Next Case";

        processAndPrintOutput(input1);
        processAndPrintOutput(input2);
    }

    public static void processAndPrintOutput(String input) {
        input = input.toLowerCase();
```

```

String vowels = "";
String consonants = "";

for (char c : input.toCharArray()) {
    if (Character.isLetter(c)) {
        if (isVowel(c)) {
            vowels += c;
        } else {
            consonants += c;
        }
    }
}

char[] vowelsArray = vowels.toCharArray();
char[] consonantsArray = consonants.toCharArray();
Arrays.sort(vowelsArray);
Arrays.sort(consonantsArray);

String sortedVowels = new String(vowelsArray);
String sortedConsonants = new String(consonantsArray);

System.out.println("Input one line of words (S) : " + input);
System.out.println("Vowel Characters : " + sortedVowels);
System.out.println("Consonant Characters : " + sortedConsonants);
System.out.println();
}

public static boolean isVowel(char c) {
    return "aeiou".indexOf(c) != -1;
}
}

```

## PSBB ( Pembatasan Sosial Berskala Besar )

### Problem

Galih and Ratna married during the COVID 19 period and only invited the families of both partners. they rented a number of minibuses to pick up all of their families to go to the wedding. But during COVID 19, the government held a PSBB program to reduce the impact of the spread of the virus. Each mini bus can only carry at most 4 passengers. What a minimum number of buses will they need to rent if all members of each family should ride in the same Busses. (one bus can't take more than two family)

## Input

The first line contains integer  $n$  — the number of families. The second line contains a sequence of integers. The integers are separated by a space, each integer is the number of members in the family.

## Output

Print the single number — the minimum number of buses necessary to drive all family to the Wedding. Print “Input must be equal with count of family” if input number of family is not equal with count of family.

## Test Case

Input	Output
Input the number of families : 5 Input the number of members in the family (separated by a space) : 1 2 4 3 3	Minimum bus required is : 4
Input the number of families : 8 Input the number of members in the family (separated by a space) : 2 3 4 4 2 1 3 1	Minimum bus required is : 5
Input the number of families : 5 Input the number of members in the family (separated by a space) : 1 5	Input must be equal with count of family

## Source Code

```
import java.util.ArrayList;
import java.util.Collections;
import java.util.List;
import java.util.Scanner;

public class PSBB {

    public static void main(String[] args) {
        try {
            Scanner scanner = new Scanner(System.in);

            System.out.print("Input the number of families: ");
            int familyCount = Integer.parseInt(scanner.nextLine());

            System.out.print("Input the number of members in the family (separated by a space): ");
            String[] membersInput = scanner.nextLine().split(" ");
            int[] members = new int[membersInput.length];
```

```

    for (int i = 0; i < membersInput.length; i++) {
        members[i] = Integer.parseInt(membersInput[i]);
    }

    if (familyCount != members.length) {
        System.out.println("Input must be equal with count of family");
        return;
    }

    int car = 0;
    List<Integer> memberList = new ArrayList<>();

    for (int member : members) {
        memberList.add(member);
    }

    // sorting data array
    Collections.sort(memberList, Collections.reverseOrder());

    // fulfill bus
    int i = 0;
    while (i < memberList.size()) {
        if (memberList.get(i) == 4) {
            car++;
            i++;
        } else {
            boolean found = false;
            for (int j = i + 1; j < memberList.size(); j++) {
                if (memberList.get(i) + memberList.get(j) <= 4) {
                    car++;
                    memberList.remove(j);
                    found = true;
                    break;
                }
            }
            if (!found) {
                car++;
            }
            i++;
        }
    }

    System.out.println("Minimum bus required is : " + car);
} catch (Exception e) {
    System.err.println("Error: " + e.getMessage());
}
}

```

}