

## CASE1

<https://www.1point3acres.com/bbs/forum.php?mod=viewthread&tid=662080&extra=page%3D1%26filter%3Dsortid%26sortid%3D311%26searchoption%5B3086%5D%5Bvalue%5D%3D10%26searchoption%5B3086%5D%5Btype%5D%3Dradio%26searchoption%5B3088%5D%5Bvalue%5D%3D1%26searchoption%5B3088%5D%5Btype%5D%3Dradio%26searchoption%5B3089%5D%5Bvalue%5D%5B5%5D%3D5%26searchoption%5B3089%5D%5Btype%5D%3Dcheckbox%26searchoption%5B3090%5D%5Bvalue%5D%3D1%26searchoption%5B3090%5D%5Btype%5D%3Dradio%26searchoption%5B3046%5D%5Bvalue%5D%3D5%26searchoption%5B3046%5D%5Btype%5D%3Dradio%26searchoption%5B3109%5D%5Bvalue%5D%3D1%26searchoption%5B3109%5D%5Btype%5D%3Dradio%26sortid%3D311%26orderby%3Ddateline>

第 1 题: 简单 dfs 找 clusters 个数, 图在下面, 当给大家省个读题时间啦

鼠标滚轮缩放图片

🔍 🖨 📄

The current selected programming language is **C++**. We emphasize the submission of a fully working code over partially correct but efficient code. Once **submitted**, you cannot review this problem again. You can use *cout* to debug your code. The *cout* may not work in case of syntax/runtime error. The version of **G++** being used is **8.1.0**.

Amazon has a 2D grid of servers. All servers are running a special software which is represented by a single lowercase letter. Adjacent servers running the same software are organized in clusters. An adjacent server is either on the left, right, above or below a given server.

As part of the daily diagnostics, a grid of all servers is printed, where each cell contains a single lowercase letter - the software that the server in that particular cell is running. Write an algorithm to find how many clusters are in the grid currently.

### Input

The input to the function/method consists of two arguments:  
*numOfRows*, an integer representing the number of rows in the grid;  
*grid*, a list of strings representing the 2D grid of servers.

### Output

Return an integer representing the number of clusters present in the grid.

### Constraints

$1 \leq \text{numOfRows} \leq 10^5$

$1 \leq \text{len} \leq 10^5$ , where len is the length of each string in the *grid*

$1 \leq \text{numOfRows} * \text{len} \leq 10^5$

亩  
@一亩三分地


第 2 题: 还是 Maximum Average Subtree, 只是背景换成亚麻内部软件更新速度测评, 平均代码量大的 subtree move fastest

鼠标滚轮缩放图片



**Problem** | **Test Cases** | **Output**

**Constraints**  
 $1 \leq \text{numOfRows} \leq 10^5$   
 $1 \leq \text{len} \leq 10^5$ , where len is the length of each string in the *grid*  
 $1 \leq \text{numOfRows} * \text{len} \leq 10^5$

 **Note**  
grid[i][j] will consist of English alphabets 'a', 'b' and 'c' only.

**Example**  
Input:  
numOfRows = 3  
grid = ["aabba", "aabba", "aaacb"]  
  
Output:  
5

**Explanation:**  
Each string represents a row of the servers and each letter represents the name of the software. The diagram below shows the 5 clusters. Note that a and b are organized in 2 clusters, and c in only one. The array grid is shown below.

**Diagnostics Map**

a	a	b	b	a
a	a	b	b	a
a	a	a	c	b

**Clusters: 5**

a	a	b	b	a
a	a	b	b	a
a	a	a	c	b

@一亩三分地

## CASE2

<https://www.1point3acres.com/bbs/forum.php?mod=viewthread&tid=661996&extra=page%3D1%26filter%3Dsortid%26sortid%3D311%26searchoption%5B3086%5D%5Bvalue%5D%3D10%26searchoption%5B3086%5D%5Btype%5D%3Dradio%26searchoption%5B3088%5D%5Bvalue%5D%3D1%26searchoption%5B3088%5D%5Btype%5D%3Dradio%26searchoption%5B3089%5D%5Bvalue%5D%5B5%5D%3D5%26searchoption%5B3089%5D%5Btype%5D%3Dcheckbox%26searchoption%5B3090%5D%5Bvalue%5D%3D1%26searchoption%5B3090%5D%5Btype%5D%3Dradio%26searchoption%5B3046%5D%5Bvalue%5D%3D5%26searchoption%5B3046%5D%5Btype%5D%3Dradio%26searchoption%5B3109%5D%5Bvalue%5D%3D1%26searchoption%5B3109%5D%5Btype%5D%3Dradio%26sortid%3D311%26orderby%3Ddateline>

昨天做的[亚麻](#) oa2 :

第一题：改回文

输入一个回文 string，改一个字母使它变成非回文，改完的 string 要比原来的 alphabetical 小，如果不行返回 impossible

第二题：item 排序

根据给的 parameter 来对产品进行排序，输出对应位置的产品名称 medium 难度 没有算法的考察 但是要会写 comparator

完全是字面意思，地里有

写 comparator 暴力秒了

## CASE3

<https://www.1point3acres.com/bbs/forum.php?mod=viewthread&tid=661899&extra=page%3D1%26filter%3D%5Btype%5D%3Dradio%26searchoption%5B3089%5D%5Bvalue%5D%5B5%5D%3D5%26searchoption%5B3089%5D%5Bvalue%5D%5B5%5D%3D5%26searchoption%5B3089%5D%5Btype%5D%3Dradio%26searchoption%5B3109%5D%5Bvalue%5D%3D1%26searchoption%5B3109%5D%5Btype%5D%3Dradio%26sortid%3D311%26orderby%3Ddateline>

热带雨林欧哎二：第一题：选组。

第二题：找最近同 x/y 坐标的城市

## CASE4 素数串

<https://www.1point3acres.com/bbs/forum.php?mod=viewthread&tid=661768&extra=page%3D1%26filter%3Dsortid%26sortid%3D311%26searchoption%5B3086%5D%5Bvalue%5D%3D10%26searchoption%5B3086%5D%5Btype%5D%3Dradio%26searchoption%5B3088%5D%5Bvalue%5D%3D1%26searchoption%5B3088%5D%5Btype%5D%3Dradio%26searchoption%5B3089%5D%5Bvalue%5D%5B5%5D%3D5%26searchoption%5B3089%5D%5Btype%5D%3Dcheckbox%26searchoption%5B3090%5D%5Bvalue%5D%3D1%26searchoption%5B3090%5D%5Btype%5D%3Dradio%26searchoption%5B3046%5D%5Bvalue%5D%3D5%26searchoption%5B3046%5D%5Btype%5D%3Dradio%26searchoption%5B3109%5D%5Bvalue%5D%3D1%26searchoption%5B3109%5D%5Btype%5D%3Dradio%26sortid%3D311%26orderby%3Ddateline>

The current selected programming language is **Java**. We emphasize the submission of a fully working code over partially correct but efficient code. Once **submitted**, you cannot review this problem again. You can use `System.out.println()` to debug your code. The `System.out.println()` may not work in case of syntax/runtime error. The version of **JDK** being used is **1.8**.

Amazon's operations team needs an algorithm that can break out a list of products for a given order. The products in the order are listed as a string and the order items are represented as prime numbers. Given a string consisting of digits [0-9], count the number of ways the given string can be split into prime numbers, which represent unique items in the order. The digits must remain in the order given and the entire string must be used.

Write an algorithm to find the number of ways the given string can be split into unique prime numbers using the entire string.

#### Input

The input to the function/method consists of an argument:  
`inputString`, a string representing the input string.

#### Output

Return an integer representing the number of ways the given string can be split into unique primes using the entire string.

#### Note

The `inputString` does not contain leading zeros.  
Each number split of the given number must be in the range 2 to  $10^6$  inclusive.  
Since the answer can be large, return the answer modulo (1000000007)

#### Constraints

$1 \leq \text{length of } inputString \leq 10^5$

#### Example

Input:  
`inputString = 11373`

Output:  
6

Explanation:  
This string can be split into prime numbers in 6 different ways: [11, 3, 7, 3], [113, 7, 3], [11, 3, 73], [11, 37, 3], [113, 73] and [11, 373].  
So the output is 6.



## CASE5

<https://www.1point3acres.com/bbs/forum.php?mod=viewthread&tid=661664&extra=page%3D1%26filter%3Dsortid%26sortid%3D311%26searchoption%5B3086%5D%5Bvalue%5D%3D10%26searchoption%5B3086%5D%5Btype%5D%3Dradio%26searchoption%5B3088%5D%5Bvalue%5D%3D1%26searchoption%5B3088%5D%5Btype%5D%3Dradio%26searchoption%5B3089%5D%5Bvalue%5D%5B5%5D%3D5%26searchoption%5B3089%5D%5Btype%5D%3Dcheckbox%26searchoption%5B3090%5D%5Bvalue%5D%3D1%26searchoption%5B3090%5D%5Btype%5D%3Dradio%26searchoption%5B3046%5D%5Bvalue%5D%3D5%26searchoption%5B3046%5D%5Btype%5D%3Dradio%26searchoption%5B3109%5D%5Bvalue%5D%3D1%26searchoption%5B3109%5D%5Btype%5D%3Dradio%26sortid%3D311%26orderby%3Ddateline>

刚做完 OA2, 补充一道地里没有的题, 时间紧张忘了照相

第二题: 产品每个 item 有三个 field, 一个 name 是 String, 另外两个是 int, 要求是可以根据不同的用户需求对不同的 field 进行排序, 0 -> name, 1 -> int1, 2 -> int2, 然后还要决定升序 (0) 还是



降序 (1), 排列后的结果要放在不同的 page 上, 每页存放结果的个数有限制, 最后根据用户的需要返回在某个 page 上的所有产品的 name

input : 一个 map, key 是 item name, value 是那两个 int fields, int field(0, 1, 2), int order (0 升, 1 降), int itemPerPage, int pageNum

output : List<String>

example : <item1, 2, 3> <item2, 3, 4> <item3, 4, 2>, field = 1, order = 1, itemPerPage = 2, pageNum = 1 结果按照中间那个 int 降序排, 应该返回 <item1>

总结: 题不难, input 一堆, 要写吐了。。。求各位路过的留点米。。

#### CASE6

<https://www.1point3acres.com/bbs/forum.php?mod=viewthread&tid=661662&extra=page%3D1%26filter%3Dsortid%26sortid%3D311%26searchoption%5B3086%5D%5Bvalue%5D%3D10%26searchoption%5B3086%5D%5Btype%5D%3Dradio%26searchoption%5B3088%5D%5Bvalue%5D%3D1%26searchoption%5B3088%5D%5Btype%5D%3Dradio%26searchoption%5B3089%5D%5Bvalue%5D%5B5%5D%3D5%26searchoption%5B3089%5D%5Btype%5D%3Dcheckbox%26searchoption%5B3090%5D%5Bvalue%5D%3D1%26searchoption%5B3090%5D%5Btype%5D%3Dradio%26searchoption%5B3046%5D%5Bvalue%5D%3D5%26searchoption%5B3046%5D%5Btype%5D%3Dradio%26searchoption%5B3109%5D%5Bvalue%5D%3D1%26searchoption%5B3109%5D%5Btype%5D%3Dradio%26sortid%3D311%26orderby%3Ddateline>

今年感觉都是新题普遍难度有所提升, 抽中两道地理见过的题, 给大家提供个 dp 吧

第一题 max average subtree

第二题 sort website items 就是给你一些网页, 让你根据相关度, 名字之类的排序然后返回网站某一页显示的东西

大家加油~

#### CASE7

<https://www.1point3acres.com/bbs/forum.php?mod=viewthread&tid=661560&extra=page%3D1%26filter%3Dsortid%26sortid%3D311%26searchoption%5B3086%5D%5Bvalue%5D%3D10%26searchoption%5B3086%5D%5Btype%5D%3Dradio%26searchoption%5B3088%5D%5Bvalue%5D%3D1%26searchoption%5B3088%5D%5Btype%5D%3Dradio%26searchoption%5B3089%5D%5Bvalue%5D%5B5%5D%3D5%26searchoption%5B3089%5D%5Btype%5D%3Dcheckbox%26searchoption%5B3090%5D%5Bvalue%5D%3D1%26searchoption%5B3090%5D%5Btype%5D%3Dradio%26searchoption%5B3046%5D%5Bvalue%5D%3D5%26searchoption%5B3046%5D%5Btype%5D%3Dradio%26searchoption%5B3109%5D%5Bvalue%5D%3D1%26searchoption%5B3109%5D%5Btype%5D%3Dradio%26sortid%3D311%26orderby%3Ddateline>

第一个蠢口 忆散而扒，但比它要简单一些。给一个回文串，替换一个字母，破坏回文，且使新串字母序最小。我的解法是从左到右找到第一个不为'a'的字母替换成'a'。需要注意的是，如果串长度是奇数，并且要替换的字符刚好在串正中间，那么即使换成'a'，新串仍然是回文，这是不成立的。如果无法替换，返回'IMPOSSIBLE'。

第二题蠢口 而散久，思路相同，从左到右滑窗，取窗口中的数字的最小值，最后再取所有最小值中的最大值。我的解法是维护一个非递减的 deque，每个窗口的最小值是 deque 的第一个元素。

## CASE8

<https://www.1point3acres.com/bbs/forum.php?mod=viewthread&tid=661507&extra=page%3D1%26filter%3Dsortid%26sortid%3D311%26searchoption%5B3086%5D%5Bvalue%5D%3D10%26searchoption%5B3086%5D%5Btype%5D%3Dradio%26searchoption%5B3088%5D%5Bvalue%5D%3D1%26searchoption%5B3088%5D%5Btype%5D%3Dradio%26searchoption%5B3089%5D%5Bvalue%5D%5B5%5D%3D5%26searchoption%5B3089%5D%5Btype%5D%3Dcheckbox%26searchoption%5B3090%5D%5Bvalue%5D%3D1%26searchoption%5B3090%5D%5Btype%5D%3Dradio%26searchoption%5B3046%5D%5Bvalue%5D%3D5%26searchoption%5B3046%5D%5Btype%5D%3Dradio%26searchoption%5B3109%5D%5Bvalue%5D%3D1%26searchoption%5B3109%5D%5Btype%5D%3Dradio%26sortid%3D311%26orderby%3Ddateline>

题目 1：根据容量拉最多的货。

题目 2：给一个数字 String, 然后求能分割成多少种纯 prime number 组合。

The current selected programming language is **Java**. We emphasize the submission of a fully working code over partially correct but efficient code. Once **submitted**, you cannot review this problem again. You can use `System.out.println()` to debug your code. The `System.out.println()` may not work in case of syntax/runtime error. The version of **JDK** being used is **1.8**.

An Amazon warehouse manager needs to create a shipment to fill a truck. All of the products in the warehouse are in boxes of the same size. Each product is packed in some number of units per box.

Given the number of boxes the truck can hold, write an algorithm to determine the maximum number of units of any mix of products that can be shipped.

#### Input

The input to the function/method consists of five arguments:

`num`, an integer representing number of products;

`boxes`, a list of integers representing the number of available boxes for products;

`unitSize`, an integer representing size of `unitsPerBox`;

`unitsPerBox`, a list of integers representing the number of units packed in each box;

`truckSize`, an integer representing the number of boxes the truck can carry.

#### Output

Return an integer representing the maximum units that can be carried by the truck.

#### Constraints

$1 \leq |\text{boxes}| \leq 10^5$

$|\text{boxes}| == |\text{unitsPerBox}|$

$1 \leq \text{boxes}[i] \leq 10^7$

$1 \leq i < |\text{boxes}|$

$1 \leq \text{unitsPerBox}[j] \leq 10^5$

$1 \leq j < |\text{unitsPerBox}|$

$1 \leq \text{truckSize} \leq 10^8$

#### Note

`|list name|` denotes length of the list.

#### Example

Input:

`num = 3`

`boxes = [1, 2, 3]`

`unitSize = 3`

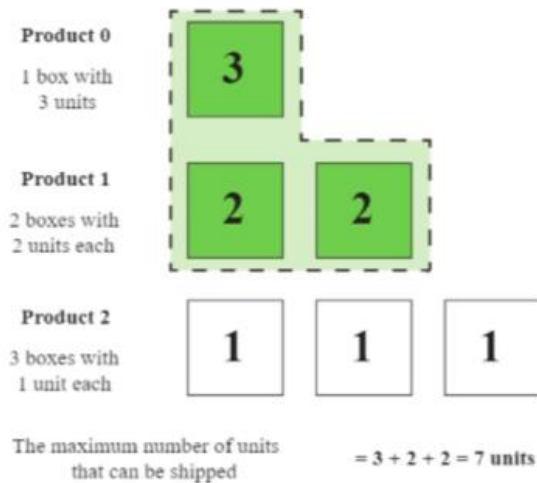
`unitsPerBox = [3, 2, 1]`

`truckSize = 3`

Output:

7

Explanation:



The maximum number of units that can be shipped is  $3 + 2 + 2 = 7$  units.

#### CASE9

<https://www.1point3acres.com/bbs/forum.php?mod=viewthread&tid=661488&extra=page%3D1%26filter%3Dsortid%26sortid%3D311%26searchoption%5B3086%5D%5Bvalue%5D%3D10%26searchoption%5B3086%5D%5Btype%5D%3Dradio%26searchoption%5B3088%5D%5Bvalue%5D%3D1%26searchoption%5B3088%5D%5Btype%5D%3Dradio%26searchoption%5B3089%5D%5Bvalue%5D%5B5%5D%3D5%26searchoption%5B3089%5D%5Btype%5D%3Dcheckbox%26searchoption%5B3090%5D%5Bvalue%5D%3D1%26searchoption%5B3090%5D%5Btype%5D%3Dradio%26searchoption%5B3046%5D%5Bvalue%5D%3D5%26searchoption%5B3046%5D%5Btype%5D%3Dradio%26searchoption%5B3109%5D%5Bvalue%5D%3D1%26searchoption%5B3109%5D%5Btype%5D%3Dradio%26sortid%3D311%26orderby%3Ddateline>

max of min route in matrix \* 3

卡车装货 \* 1

最大平均子竖 \* 2

素数串 \* 1

#### CASE10 专家打分

<https://www.1point3acres.com/bbs/forum.php?mod=viewthread&tid=661422&extra=page%3D1%26filter%3Dsortid%26sortid%3D311%26searchoption%5B3086%5D%5Bvalue%5D%3D10%26searchoption%5B3086%5D%5Btype%5D%3Dradio%26searchoption%5B3088%5D%5Bvalue%5D%3D1%26searchoption%5B3088%5D%5Btype%5D%3Dradio%26searchoption%5B3089%5D%5Bvalue%5D%5B5%5D%3D5%26searchoption%5B3089%5D%5Btype%5D%3Dcheckbox%26searchoption%5B3090%5D%5Bvalue%5D%3D1%26searchoption%5B3090%5D%5Btype%5D%3Dradio%26searchoption%5B3046%5D%5Bvalue%5D%3D5%26searchoption%5B3046%5D%5Btype%5D%3Dradio%26searchoption%5B3109%5D%5Bvalue%5D%3D1%26searchoption%5B3109%5D%5Btype%5D%3Dradio%26sortid%3D311%26orderby%3Ddateline>



a fully working code over partially correct but efficient code. Once **submitted**, you cannot review this problem again. You can use `System.out.println()` to debug your code. The `System.out.println()` may not work in case of syntax/runtime error. The version of **JDK** being used is **1.8**.

An Amazon Area Manager is trying to assemble a specialized team from a roster of available associates. There is a minimum number of associates to be involved, and each associate needs to have a skill rating within a certain range. Given a list of associates' skill levels with desired upper and lower bounds, determine how many teams can be created from the list.

Write an algorithm to find the number of teams that can be created fulfilling the criteria.

#### Input

The input to the function/method consists of five arguments:

*num*, an integer representing the number of associates;

*skills*, a list of integers representing the skill levels of associates;

*minAssociates*, an integer representing the minimum number of team members required;

*minLevel*, an integer representing the lower limit for skill level, inclusive;

*maxLevel*, an integer representing the upper limit for skill level, inclusive.

#### Output

Return an integer representing the total number of teams that can be formed per the criteria.

#### Constraints

$$1 \leq num \leq 20$$

$$1 \leq minAssociates \leq num$$

$$1 \leq minLevel \leq maxLevel \leq 1000$$

$$1 \leq skills[i] \leq 1000$$

$$0 \leq i < num$$

#### Example

Input:

*num* = 6

*skills* = [12, 4, 6, 13, 5, 10]

*minAssociates* = 3

*minLevel* = 4

*maxLevel* = 10

Output:

5

Explanation:

The list includes associates with skill levels [12, 4, 6, 13, 5, 10]. They want to hire at least 3 associates with skill levels between 4 and 10, inclusive. Four of the associates with the following skill levels {4, 6, 5, 10} meet the criteria. There are 5 ways to form a team of 3 associates : {4, 5, 6}, {4, 6, 10}, {4, 5, 10}, {5, 6, 10}, and {4, 5, 6, 10}.

So the output is 5.

## CASE11 网页 item

鼠标滚轮缩放图片



### Problem | Test Cases | Output

The current selected programming language is **Java**. We emphasize the submission of a fully working code over partially correct but efficient code. Once **submitted**, you cannot review this problem again. You can use `System.out.println()` to debug your code. The `System.out.println()` may not work in case of syntax/runtime error. The version of **JDK** being used is **1.8**.

Amazon's website contains one to many items in each page. To mimic the logic of the website, an Amazon programmer has a list of items and each item has its name, relevance and price. After sorting the items by (name: 0, relevance: 1, price: 2), the programmer is trying to find out a list of items displayed in a chosen page.

Given a list of items, the sort column, the sort order (0: ascending, 1: descending), the number of items to be displayed in each page and a page number, write an algorithm to determine the list of item names in the specified page while respecting the item's order (Page number starts at 0).

#### Input

The input to the function/method consists of six arguments:

*numOfItems*, an integer representing the number of items;

*items*, a map of string as key representing the name and pair of integers as values representing the relevance, price.

*sortParameter*, an integer representing the value used for sorting (0 for name, 1 for relevance, 2 for price)

*sortOrder*, an integer representing the order of sorting (0 for ascending order and 1 for descending order);



### Output

Return a list of strings representing the item names on the requested page in the order they are displayed.

### Constraints

$$1 \leq \text{numOfItems} < 10^5$$

$$0 \leq \text{relevance, price} < 10^8$$

$$0 \leq \text{pageNumber} < 10$$

### Note

*itemsPerPage* is always greater than 0, and is always less than the minimum of *numOfItems* and 20.

### Example

Input:

*numOfItems* = 3

*items* = [{"item1", 10, 15}, {"item2", 3, 4}, {"item3", 17, 8}]

*sortParameter* = 1

*sortOrder* = 0

*itemsPerPage* = 2

*pageNumber* = 1

Output:

["item3"]

Explanation:

There are 3 items.

Sort them by relevance(*sortParameter* = 1) in ascending order ( *items* = [{"item2", 3, 4}, {"item1", 10, 15}, {"item3", 17, 8}] ).

Display up to 2 items on each page.

The page 0 contains 2 item names ["item2", "item1"] and page 1 contains only 1 item name ["item3"].

So, the output is "item3".

亩

@一亩三分地

亩

## CASE12 最近城市

Amazon has Fulfillment Centers in multiple cities within a large geographic region. The cities are arranged on a graph that has been divided up like an ordinary Cartesian plane. Each city is located at an integral  $(x, y)$  coordinate intersection. City names and locations are given in the form of three arrays:  $c$ ,  $x$ , and  $y$ , which are aligned by the index to provide the city name  $c[i]$ , and its coordinates,  $(x[i], y[i])$ .

Write an algorithm to determine the name of the nearest city that shares either an  $x$  or a  $y$  coordinate with the queried city. If no other cities share an  $x$  or  $y$  coordinate, return `NONE`. If two cities have the same distance to the queried city,  $q[i]$ , consider the one with an alphabetically smaller name (i.e. `'ab' < 'aba' < 'abb'`) as the closest choice.

The distance is denoted on a Euclidean plane: the difference in  $x$  plus the difference in  $y$ .

### Input

The input to the function/method consists of six arguments:

*numOfCities*, an integer representing the number of cities;

*cities*, a list of strings representing the names of each city[i];

*xCoordinates*, a list of integers representing the X coordinates of each city[i];

*yCoordinates*, a list of integers representing the Y-coordinates of each city[i];

*numOfQueries*, an integer representing the number of queries;

*queries*, a list of strings representing the names of the queried cities.

### Output

Return a list of strings representing the name of the nearest city that shares either an  $x$  or a  $y$  coordinate with the queried city.

### Constraints

$1 \leq \text{numOfCities}, \text{numOfQueries} \leq 10^5$

$1 \leq xCoordinates[i], yCoordinates[i] \leq 10^9$

$1 \leq \text{length of queries[i] and cities[i]} \leq 10$



## Constraints

$1 \leq \text{numOfCities}, \text{numOfQueries} \leq 10^5$   
 $1 \leq \text{xCoordinates}[i], \text{yCoordinates}[i] \leq 10^9$   
 $1 \leq \text{length of queries}[i] \text{ and cities}[i] \leq 10$

## Note

Each character of all  $c[i]$  and  $q[i]$  is in the range `ascii[a-z, 0-9, -]`.  
 All city name values,  $c[i]$ , are unique. All cities have unique coordinates.

## Examples

Example 1:

Input:

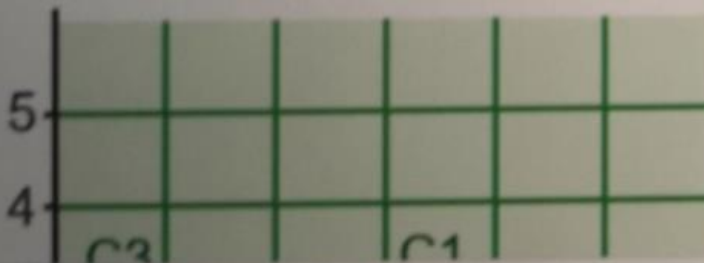
```

numOfCities = 3
cities = ["c1", "c2", "c3"]
xCoordinates = [3, 2, 1]
yCoordinates = [3, 2, 3]
numOfQueries = 3
queries = ["c1", "c2", "c3"]
    
```

Output:

```
[c3, NONE, c1]
```

Explanation:





## CASE13 债务最多的人

### Problem | Test Cases | Output |

The current selected programming language is **Java**. We emphasize the submission of a fully working code over partially correct but efficient code. Once **submitted**, you cannot review this problem again. You can use `System.out.println()` to debug your code. The `System.out.println()` may not work in case of syntax/runtime error. The version of **JDK** being used is **1.8**.

Amazon is working on a new application for recording internal debts across teams. This program can be used to create groups that show all records of debts between the group members. Given the group debt records observed for this team (including the borrower name, lender name, and debt amount), who in the group has the smallest negative balance?

Notes: -10 is smaller than -1. If multiple people have the smallest negative balance, return the list in alphabetical order. If nobody has a negative balance, return the list consisting of string "Nobody has a negative balance".

Write an algorithm to find who in the group has the smallest negative balance.

#### Input

The input to the function/method consists of three arguments:

Problem | Test Cases | Output |

borrower	lender	amount
Alex	Blake	2
Blake	Alex	2
Casey	Alex	5
Blake	Casey	7
Alex	Blake	4
Alex	Casey	4

The first, fifth, and sixth entries decrease Alex's balance because Alex is a borrower in them. The second and third entries increase it because Alex is a lender. So, Alex's balance is  $(2+5) - (2+4+4) = 7 - 10 = -3$ . Blake is a lender in the first and fifth entries and a borrower in the second and fourth entries. Thus, Blake's balance is  $(2+4) - (2+7) = 6 - 9 = -3$ . Casey is a lender in the third entry and a

Problem | Test Cases | Output |

balance of -3, which is the minimum among all members.

#### Helper Description

The following class is used to represent an `debtRecord` and is already implemented in the default code (Do not write this definition again in your code):

```
public class debtRecord
{
    String borrower = "";
    String lender = "";
    int amount = 0;
    debtRecord()
    {
        // empty constructor
    }
    debtRecord(String borrower, String lender, int amount)
    {
        this.borrower = borrower;
        this.lender = lender;
        this.amount = amount;
    }
}
```

## CASE14 选 tenure 最大的组

You are required to implement the algorithm in **Python** only. We emphasize the submission of a fully working code over partially correct but efficient code. Use of certain header files are restricted. Once **submitted**, you cannot review this problem again. You can use *print* to debug your code. The version of Python being used is **3.5**

Imagine that an employment tree represents the formal employee hierarchy at Amazon. Manager nodes have child nodes for each employee that reports to them; each of these employees can, in turn, have child nodes representing their respective reportees. Each node in the tree contains an integer representing the number of months the employee has spent at the company. Team tenure is computed as the average tenure of the manager and all the company employees working below the manager. The oldest team has the highest team tenure.

Write an algorithm to find the manager of the team with the highest tenure. An employee must have child nodes to be a manager.

### Input

The input to the function/method consists of an argument - *president*, a node representing the root node of the employee hierarchy.

### Output

Return the node which has the oldest team.

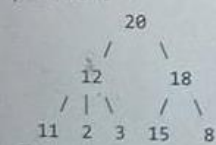
### Note

There will be at least one child node in the tree and there will be no ties.

### Example

Input:

*president* =



Output:

**Output**

Return the node which has the oldest team.

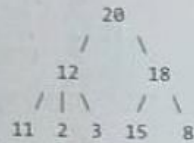
**Note**

There will be at least one child node in the tree and there will be no ties.

**Example**

Input:

president =



Output:

18

Explanation:

There are three managers in this tree with the following team tenures:

$$12 \Rightarrow (11+2+3+12)/4 = 7$$

$$18 \Rightarrow (18+15+8)/3 = 13.67$$

$$20 \Rightarrow (12+11+2+3+18+15+8+20)/8 = 11.125$$

The oldest team is the team of the manager with 18 months at the company.  
So, the output is 18.

**Helper Description**

The following class is used to represent a EmployeeNode and is already implemented in the default code (Do not write this definition again in your code):

```

class EmployeeNode:
    def __init__(self, numOfCalls):
        self.value = numOfCalls # value represents the number of calls made to this node
        self.children = [] # children represents the children nodes of this node
  
```

**CASE15**

Your team at Amazon is overseeing the design of a new, high-efficiency data center at HQ2. A power grid needs to be generated for supplying power to N servers. All servers in the grid have to be connected such that they have access to power. The cost of connections between different servers varies.

Assume that there are no ties, the names of servers are unique, connections are directionless, there is at most one connection between a pair of servers, all costs are greater than zero, and a server does not connect to itself.

Write an algorithm to minimize the cost of connecting all servers in the power grid.

**Input**

The input to the function/method consists of two arguments -

*num*, an integer representing the number of connections.

*connection*, representing a list of Connections where each element of the list consists of two servers and the cost of connection between the servers.

**Output**

Return a list of Connections where each element of the list consists of two servers and the cost of connection between the servers such that all servers are connected at the lowest total cost. If no such Connections exists, then return a list with empty Connection.

**Note**

The cost of connection between the servers is always greater than 0.

**Example**

Input:

*num* = 5

*connection* =

[[A, B, 1],



between the servers such that all servers are connected at the lowest total cost. If no such connections exist, then return a list with empty Connection.

#### Note

The cost of connection between the servers is always greater than 0.

#### Example

Input:

num = 5

connection =

[[A, B, 1],

[B, C, 4],

[B, D, 6],

[D, E, 5],

[C, E, 1]]

Output:

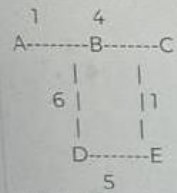
[[A, B, 1],

[B, C, 4],

[C, E, 1],

[D, E, 5]]

Explanation:



By connecting A to B, B to C, C to E, and E to D, the cost of connection is minimum.  
So, the output is

[[A, B, 1],

[B, C, 4],

By connecting A to B, B to C, C to E, and E to D, the cost of connection is minimum.  
So, the output is

[[A, B, 1],

[B, C, 4],

[C, E, 1],

[D, E, 5]]

#### Helper Description

The following class is used to represent a connection which is already implemented in the default code (Do not write this definition again in your code):

```
class Connection:
    def __init__(self, one, two, cost):
        self.firstTown = one
        self.secondTown = two
        self.cost = cost
```