对于解决方案，我们要求您使用Java、Ruby、c#、Python、Clojure、Scala或JavaScript

必须有一种方法通过文本文件向应用程序提供输入数据；

* The application must run

您应该提供充分的证据，证明您的解决方案是完整的，说明它与所提供的测试数据是正确的。

请使用此邮件下方的URL提交您的代码，同时请将作业附在您的招聘人员的邮件中。

**Rules:**

您可能不会使用任何外部库来解决这个问题，但是您可以使用外部库或工具来构建或测试。具体地说，您可以使用单元测试库或为所选语言可用的构建工具(例如，JUnit、Ant、NUnit、Rspec、Rake等)。

系统安全性对我们非常重要，为了安全目的，某些文件扩展将被阻塞，从而导致应用程序的延迟。

不应该包含任何可执行附件，包括带有.exe或.lib扩展名的附件。

我们需要能够自己运行和构建您的代码，因此请将您的代码作为压缩文件提交给源代码和支持文件，而不需要任何编译代码。

如果您提交c#，请不要提交您的代码作为。msi文件。

请包含您的设计和假设的简要说明，以及您的代码，以及运行应用程序的详细说明。

我们将评估许多事情，包括解决方案的设计方面和面向对象编程技能。

虽然这些都是小问题，但我们希望您提交您认为是产品质量的代码;

可以运行、维护和演进的代码。

你不需要为你的解决方案镀金，但是我们正在寻找的不仅仅是一个简单的算法。

我们希望我们的招聘过程公平，让每个人都从同一个地方开始。

为了实现这一点，我们要求您不要共享或发布这些问题。

上传前请将文件压缩成一个.zip文件。

请确保您的提交中没有可执行文件。

我们的系统出于安全目的阻止可执行文件，我们希望避免您的进程中出现任何延迟。

\*\*可执行文件包括asp、bat、class、cmd、com、cpl、dll、exe、fon、hta、ini、ins、iw、jar、jsp、js、jse、pif、scr、shs、sh、vb、vbe、vbs、ws、wsc、wsf、wsh & msi。

*一般来说，我们允许你在收到这些指示后的三天内提交你的代码，但是如果需要的话，你可以要求招聘人员给你更多的时间。*

# ****Problem one: Trains****

当地的通勤铁路服务于基威兰的许多城镇。

由于对货币的担忧，所有的轨道都是“单向的”。

也就是说，从Kaitaia到Invercargill的路线并不意味着存在一条从Invercargill到Kaitaia的路线。

事实上，即使这两条路线同时存在，它们也是不同的，不一定是相同的距离!

这个问题的目的是帮助铁路公司向客户提供有关线路的信息。

特别地，你将计算沿着某条路线的距离，两个城镇之间不同的路线的数量，以及两个城镇之间最短的路线。

## Input:

有向图，其中节点表示城镇，边缘表示两个城镇之间的路线。

边缘的权重表示两个城镇之间的距离。

给定的路线不会出现不止一次，而对于给定的路线，起点和终点都不会是同一个城镇。

## Output:

对于测试输入1到5，如果不存在这样的路由，则输出“不存在这样的路由”。

否则，按照所给的路线走;

不要做任何额外的停止!

例如，第一个问题是指从A城市出发，然后直接去B城市(距离为5)，然后直接去C城市(距离为4)。

A-B-C路线的距离。

A-D路线的距离。

航路A-D-C的距离。

路线A-E-B-C-D的距离。

路线A-E-D的距离。

从C点开始，到C点结束，最多3站。

在下面的示例数据中，有两个这样的行程:C-D-C(2站)。

和C-E-B-C(3站)。

从A点开始到C点结束的旅程有4站。

在下面的样本数据中，有三次这样的旅行:A到C(通过B、C、D);

A至C(通过D、C、D);

A到C(通过D E B)

从A到C的最短路径的长度(以旅行距离计算)。

从B到B的最短路径的长度(以旅行距离计算)。

距离小于30的C到C的不同路径数。

在样本数据中，trips分别为:CDC、CEBC、CEBCDC、CDCEBC、CDEBC、CEBCEBC、CEBCEBCEBC。

## Test Input:

对于测试输入，用字母A到d的前几个字母命名城镇(A到B)，距离为5的两个城镇之间的路径表示为AB5。

图:AB5, BC4, CD8, DC8, DE6, AD5, CE2, EB3, AE7

**Expected Output:**

Output #1: 9

Output #2: 5

Output #3: 13

Output #4: 22

Output #5: NO SUCH ROUTE

Output #6: 2

Output #7: 3

Output #8: 9

Output #9: 9

Output #10: 7

# **Problem Two: Conference Track Management 会议跟踪管理**

您正在计划一个大型的编程会议，并且已经收到了许多建议，这些建议已经通过了最初的屏幕过程，但是您在将它们与当天的时间限制相匹配时遇到了困难——有太多的可能性!

所以你要写一个程序来帮你。

会议有多个轨道，每个轨道都有上午和下午的会议。

每个会议包含多个会谈。

上午的课程从早上9点开始，中午12点前结束，午餐时间。

下午的会议从下午1点开始，必须在社交活动之前结束。

社交活动不能早于4点开始，也不能晚于5点开始。

没有谈话的标题有数字。

所有的谈话时间都是分钟(不是小时)或闪电(5分钟)。

主持人会非常准时;

会话之间不需要间隔。

**请注意**，根据您选择如何完成这个问题，您的解决方案可能会给出不同的顺序，或者将对话组合成轨迹。

这是可以接受的;

您不需要完全复制这里给出的示例输出。

## Test Input:

针对企业Rails 60min编写快速测试

在Python 45min中做得过多

Lua用于大众30分钟

宝石版本45min出现的Ruby错误

常见的Ruby错误45分钟

用于Python开发人员的Rails。

沟通在距离60分钟

Accounting-Driven发展45分钟

哇30分钟

坐下来写30分钟

对编程vs噪声45min

Rails魔法60分钟

Ruby on Rails:为什么我们要在60min内前进

Clojure Scala(在我的项目上)45分钟

在西雅图的荒野上编程30分钟

Ruby vs. Clojure用于后端开发30min

Ruby on Rails遗留应用程序维护60min

一个没有黑客新闻的世界

用户界面CSS在Rails应用30min

## Test output:

### Track1:

09:00正在编写针对企业Rails 60min的快速测试。

10:00在Python 45min中做得太过了。

10:45AM Lua代表群众30分钟

11:15AM Ruby错误来自于错误匹配的Gem版本45min

旧的午餐

01:00PM Ruby on Rails:为什么我们应该移动60min。

常见的Ruby错误45分钟。

对编程vs噪音45分钟

3点30分在西雅图的荒野进行节目

Ruby对Clojure的后端开发时间为30分钟

用户界面CSS在Rails应用中30分钟

05:00PM社交活动

### Track2:

09:00AM远距离通信60min

十时Rails魔法60分钟

上午11点哇30分钟

上午11:30坐下来写30min。

旧的午餐

01:00PM Accounting-Driven开发45分钟

01:45分

一个没有黑客新闻的世界

Ruby on Rails遗留应用维护60分钟

Python开发人员的Rails闪电

05:00PM社交活动

# ****Problem Three: Merchant's Guide to the Galaxy 问题三:《银河商人指南》****

在最近的金融危机让地球上99.99%的人口拥有0.01%的财富后，你们决定放弃地球。

幸运的是，你的账户里只剩下那么一点点钱，你可以租一艘宇宙飞船，离开地球，飞遍整个银河系去卖普通的金属和泥土(显然很值钱)。

在银河系买卖需要你转换数字和单位，你决定写一个程序来帮助你。

用于星系间交易的数字与罗马数字类似，你辛苦地收集了它们之间的适当翻译。

罗马数字基于七个符号:

Symbol Value

I 1

V 5

X 10

L 50

C 100

D 500

M 1,000

数字是通过把符号组合在一起并加上数值而形成的。

例如，MMVI 是 1000 + 1000 + 5 + 1 = 2006。

一般来说，符号按照价值的顺序排列，从最大的值开始。

当较小的值先于较大的值时，较小的值将从较大的值中减去，并将结果添加到总数中。

例如MCMXLIV = 1000 + (1000 − 100) + (50 − 10) + (5 − 1) = 1944。

1. 符号“I”、“X”、“C”和“M”可以连续重复三次，但不能再多。

(如果第三和第四项被较小的值(如XXXIX)分隔，它们可能出现四次。)

“D”、“L”和“V”永远不会重复。

2. “I”只能从“V”和“X”中减去。

“X” 只能从“L”和“C”中减去。

“C”只能从“D”和“M”中减去。

“V”、“L”和“D”不能被减去。

3. 任何大值符号只能减去一个小值符号。

4. 用阿拉伯数字[16]编写的数字可以被分解为数字。

例如，1903年由1,9,0和3组成。

要写出罗马数字，每一个非零的数字都应该分别处理。

在上面的示例中，1000 = M, 900 = CM, 3 = III。

因此,1903 = MCMIII。

资料来源:维基百科([17]<http://en.wikipedia.org/wiki/Roman_numerals>

程序的输入由几行文本组成，这些文本详细说明了星系间单位和罗马数字之间的转换。

您需要适当地处理无效查询。

## Test input:

glob is I

prok is V

pish is X

tegj is L

glob glob Silver is 34 Credits

glob prok Gold is 57800 Credits

pish pish Iron is 3910 Credits

how much is pish tegj glob glob ?

how many Credits is glob prok Silver ?

how many Credits is glob prok Gold ?

how many Credits is glob prok Iron ?

how much wood could a woodchuck chuck if a woodchuck could chuck wood ?

## Test output

pish tegj glob glob is 42

glob prok Silver is 68 Credits

glob prok Gold is 57800 Credits

glob prok Iron is 782 Credits

I have no idea what you are talking about

Please submit here:  
<https://app.greenhouse.io/tests/08df6e84bc48c5bc7a05f20d6b9d42b5>