评委一评分，签名及备注	队号： 10463	评委三评分，签名及备注
评委二评分，签名及备注	选题： C	评委四评分，签名及备注

题目：家庭暑期旅游最佳线路套餐设计

摘　要

现今，随着人们生活水平的提高，越来越多的家长会选择暑期带孩子去某城市去旅游。但不同的家庭有不同的需求，如在套餐费用、参观景点个数、时间、家庭人数、天气方面等的需求。因此在本文中，我们选择一个旅游城市（济南）作为代表，综合考虑影响不同家庭的出游因素，为有不同需求的家庭设计一份最佳旅游套餐。

我们通过使用控制变量法并参考人类心理学，选出典型需求进行组合与分析，建立了五个模型。其中，套餐费用、参观景点数目和时间被视为三个基本需求，我们在模型一至模型三中考虑了他们之间的关系，并在此基础上，建立了模型四与模型五，优化分析了家庭人数和天气需求的影响。

在模型一、二、三中，我们针对不同的家庭需求：1、套餐费用不限制，游览全部景点，同时花费最少的时间。2、套餐费用有具体的限制，有具体的时间限制，游览最多的景点。3、套餐费用最少的前提下，游览景点个数和时间因素进行了组合分析。在满足这些需求的条件下，我们可以选择路径最短的路线或使用较快的交通工具。然后我们用 lingo 软件处理了相关数据，并采用贪婪法和图论法进行分析，建立出费用-时间-景点数“三位一体”的需求模型，设计了较优的路线，以满足家庭的需求。

模型四、五在以上需求模型的基础上，针对多组家庭同时出游的情况和阴雨天气对旅客满意度的影响，建立新的模型来进一步减少旅行费用，以提升游客出行的满意度。因此，我们制定了结伴游的优惠策略并定义了阴雨天气的损失度，以费用最低和损失最小为目标，求出相应的解决方案。

关键字：旅游路线设计　贪婪法　图论法　需求模型　满意度

家庭暑期旅游最佳线路套餐设计

一、问题重述

暑假即将来临，很多家长会选择这个时间带孩子去某城市去旅游。但不同的家庭有不同的需求，如在套餐费、参观景点个数、时间、家庭人数和天气方面的需求。选择一个旅游城市，综合考虑不同家庭在套餐费用，参观景点个数、时间、家庭人数和天气方面的需求，为有不同需求的家庭设计一份最佳旅游套餐。

二、问题分析

根据对题目的理解，问题的求解是在满足不同家庭需求的情况下，设计一份最佳旅游套餐，从而使家庭旅游花费最少，或使用的时间最短，或游览的景点数最多，或满足其他的需求。综合考虑多种因素，我们将家庭的需求划分为以下几个方面：套餐费用，参观景点数目，时间，家庭人数，天气五个方面。

通过使用控制变量法并参考人类心理学，选出典型需求组合进行分析。其中，套餐费用、参观景点数目和时间被视为三个基本需求。模型一至模型三考虑三个基本需求之间的关系。模型四考虑家庭人数需求。模型五考虑天气需求。

本篇论文以山东济南作为旅游城市，综合考虑多种因素，进行建模分析。（济南旅游区状况见下图）

模型一针对家庭需求：套餐费用不限制，游览全部景点，同时花费最少的时间。在满足这些需求的条件下，我们可以选择路径最短的路线或使用较快的交通工具等。通过分析我们使用 Lingo 软件，设计较优的路线，以满足家庭的需求。

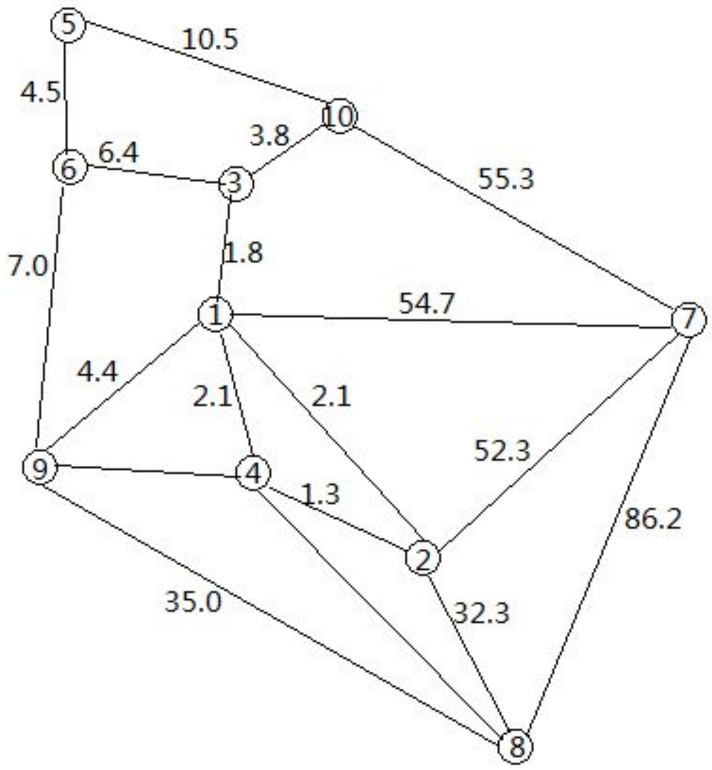
模型二针对家庭需求：套餐费用有具体的限制，有具体的时间限制，游览最多的景点。由于此处有两个限制因素，即套餐费用限制和时间限制。因此，首先分析单个因素限制，最后综合进行两个因素限制的分析。第一个分析涉及到控制时间变量，即套餐费用具体限制，游览最多的景点，时间无限制。需要建立目标函数与条件约束公式。第二个分析涉及到控制费用变量，即套餐费用不限制，游览最多的景点，有具体的时间限制。同样的需要建立目标函数与条件约束公式。最后综合第一个和第二个分析，考虑控制两个变量，得出最终的目标表达式。结合贪婪法和图论法，这样我们可以设计一条较优的路线，在套餐费用和时间限制的条件下，使游览的景点最多。

模型三针对家庭需求：套餐费用最少的前提下，游览景点个数和时间因素进行组合分析。首先分析套餐费用最少，有具体的时间限制，游览最多景点个数，的情况。做法为在满足相应的约束条件下，先确定游览的景点数，然后计算出在这种情况下的最小花费。这样最终会得出几种最佳方案，而家庭可以根据自己的实际情况进行选择。第二步分析套餐费用最少，游览全部景点个数，时间不限。这种需求组合的分析可以建立在第一个分析的基础上，仅仅是改变了时间的约束，即家庭要浏览所有的景点，因此可以使用第一个分析建立的模型。

模型四针对家庭人数需求。当两个家庭选择共同出行旅游以减少费用，而在出游时间上不能完全重合时，需要建立一个最优惠的旅游套餐。由于假设中提到

参观景点的人数越多，每人承担的费用越少。因此我们应该考虑使两组同时在外旅游时尽量在同一景点游览，来减少旅游总费用。基于此思想建立模型求解即可。

模型五针对天气需求，考虑了天气的因素。由于阴雨天气会给家庭带来一定的满意度的损失，因此，该模型考虑使阴雨天气造成的损失最小。我们在定义这个损失后，对总费用和损失两个目标分别加权，以最小为目标求出相应的方案。



1——趵突泉公园+泉城广场　2——千佛山　3——大明湖　4——泉城公园
5——黄河森林公园　6——济南动物园　7——朱家峪　8——金象山乐园
9——英雄山烈士陵园　10——洪家楼广场

三、模型假设

1. 旅游过程中无任何意外状况发生。
2. 假定暑期游客游览最长时间为两个月。
3. 各景点接待游客的能力充分高，即可以实现多条旅游路线的游客同时游览同一个景点。
4. 参观景点的人数越多，每人承担的费用越少。
5. 游客们所乘坐的旅游大巴平均时速为 50km/h，平均费用为 0.3 元/km；
6. 一个景点直接到达另外一个景点是指，途中经过的其他景点只是一个转站地，而并不进行游览；
7. 以泉城广场作为旅游的起点和终点，不存在中途离开的游客，并且游客在游览过程中，不存在经过景点而不进入景点游览的情况。

四、符号说明

T：旅客旅游的总时间；

p_1：旅客在路上所花费的时间；

p_2：旅客在景点停留的时间；

p_3：旅客可能花费的住宿时间；

e_i：表示旅客从第i个景点可能住宿时间；

n：旅游景点个数；

r_{ij}：旅客是否从第i个景点到第j个景点的 0---1 变量；

t_{ij}：旅客从第i个景点到第j个景点在路上所用的时间；

c_{ij}：表示从第i个景点到第j个景点所需的交通费用；

m：旅客的旅游总花费；

m_1：旅客的交通总费用；

m_2：旅客的旅游景点的花费；

m_3：旅客可能花费的费用（住宿费、吃饭的费用等）；

$\lambda_i^{'}$：第i个旅游景点对于第一个家庭的权重；

$\lambda_i^{''}$：第i个旅游景点对于第二个家庭的权重；

$m_1^{'}$：第一个家庭的交通总费用；

$m_1^{''}$：第二个家庭的交通总费用；

$m_2^{'}$：第一个家庭旅游景点的花费；

$m_2^{''}$：第二个家庭旅游景点的花费；

m_3：两个家庭同时在一景点旅游比分别旅游节约的费用；

α_i：两个家庭是否同时在第i个景点旅游的 0-1 变量；

P_{is}：第i个景点在第s天阴雨的概率；（s=1，2，……10）

T_i：旅客们到达第i个景点的时间；

V：旅客们游览的景点的集合，如$V=\{1，8，5，7\}$表示旅客们游览了第 1、8、5、7 个景点。

五、模型的建立与求解

根据题目要求，我们在设计家庭暑期旅游套餐时，综合考虑了旅行路线,时间，人数，天气等状况，逐步构建模型，建立一套最全，最优化的旅行套餐,以适应不同家庭的需求。

5.1 模型一的建立与求解

5.1.1 目标函数的确立：

首先，我们考虑在旅游费用不限的情况下，要求游览所有景点，使时间最少。根据我们对题目的理解，我们认为旅客的使用时间由三部分组成,即在路上的时间，在景点停留的时间和可能住宿时间。所以我们定义：

T ---旅客使用的总时间；

p_1 ---旅客在路上所花费的时间；

p_2 ---旅客在景点停留的时间；

p_3 ---旅客可能花费的住宿时间；

$\mathrm{e_i}$ ---表示旅客在第i个景点可能住宿时间；

n ---旅游景点个数。

综上所述，我们得到总的目标函数为：

$$MinT = p_1 + p_2 + p_3$$

（1）旅客在路上花费的时间：

因为t_{ij}表示旅客从第i个景点到第j个景点在路上所用的时间;用r_{ij}表示旅客是否从第i个景点到第j个景点的 0---1 变量，由于旅客游览十个景点，且是一次巡回，所以我们的旅客在路上所用时间为：

$$p_1 = \sum_{i=1}^{11}\sum_{j=1}^{11} r_{\mathrm{ij}} \times t_{\mathrm{ij}}$$

（2）旅客在景点停留的时间：

我们用t_i表示旅客在第i个景点的停留时间;用r_{ij}表示旅客是否从第i个景点

到第 j 个景点的 0---1 变量，由于旅客游览十个景点，且是一次巡回，所以我们得旅客在景点停留的时间为：

$$p_2 = \frac{1}{2}\sum_{i=1}^{11}\sum_{j=1}^{11} r_{ij}(t_i + t_j)$$

（3）旅客可能住宿时间：

用 e_i 表示旅客在第 i 个景点可能住宿时间；用 r_{ij} 表示旅客是否从第 i 个景点直接到达第 j 个景点的 0---1 变量，由于旅客游览十个景点，且是一次巡回，所以我们得旅客可能住宿时间为：

$$p_3 = \frac{1}{2}\sum_{i=1}^{11}\sum_{j=1}^{11} r_{\mathrm{ij}}(e_i + e_j)$$

综上所述，我们的总的目标函数为：

$$MinT = p_1 + p_2 + p_3$$

$$= \sum_{i=1}^{11}\sum_{j=1}^{11} r_{ij} \times t_{ij} + \frac{1}{2}\sum_{i=1}^{11}\sum_{j=1}^{11} r_{ij}(t_i + t_j) + \frac{1}{2}\sum_{i=1}^{11}\sum_{j=1}^{11} r_{ij}(e_i + e_j)$$

5.1.2 约束条件：

①旅游景点的约束

根据题目要求及假设情况，我们用 $\sum_{i=1}^{11}\sum_{j=1}^{11} r_{ij}$ 表示旅游的景点数，则我们假设旅游的景点数为 $n(n = 2,3,4,5,6,7,8,9,10,11)$，因为旅客要游览十个景点，所以 $n = 11$ 。故旅游景点数约束为：

$$\sum_{i=1}^{11}\sum_{j=1}^{11} r_{ij} = 11$$

②0—1 变量约束[1]

为了使旅游时间最少，则我们需要选择不同的旅游路线，因为本题为环形路线，且是一次巡回，所以我们可以把所有的景点连成一个圈，而把每一个景点看做圈上一个点。对于每个点来说，只允许最多一条边进入，同样只允许最多一条边出来，并且只要有一条边进入就要有一条边出去。因此可得约束：

$$\sum_{i}^{11} r_{ij} \le \sum_{j}^{11} r_{ij} \le 1 \quad (i, j = 1,2,3\ldots\ldots 9,10,11)$$

当$i=1$时，因为泉城广场是出发点，所以 $\sum_{i=1} r_{ij}=1$；

当$j=1$时，因为游客们最终要回到出发点，所以$\sum_{j=1} r_{ij}=1$；.

根据题目所述，我们可以得到以下结论：

$$\sum_{i} r_{ij}=\sum_{j} r_{ij}\le 1$$

$$\sum_{i=1} r_{ij}=1$$

$$\sum_{j=1} r_{ij}=1$$

$$(i,j=1,2,3\ldots\ldots 9,10,11)$$

同样，当$i,j\ge 2$时，根据题意不可能出现$r_{ij}=r_{ji}=1$，即不可能出现游客在两地间往返旅游，因为本题为一次巡回，则综上所述，我们可得约束为：

$$r_{ij}\times r_{ji}=0$$

5.1.3 模型建立：

根据对题目的理解，我们可以总的模型为：

$$MinT=p_1+p_2+p_3$$

$$=\sum_{i=1}^{11}\sum_{j=1}^{11} r_{ij}\times t_{ji}+\frac{1}{2}\sum_{i=1}^{11}\sum_{j=1}^{11} r_{ij}(t_i+t_j)+\frac{1}{2}\sum_{i=1}^{11}\sum_{j=1}^{11} r_{ij}(e_i+e_j)$$

约束条件：

$$\begin{cases}\sum_{i=1}^{11}\sum_{j=1}^{11} r_{ij}=11 & (i,j=1,2,3\ldots\ldots 9,10,11)\\ \sum_{i=1} r_{ij}=1,\quad \sum_{j=1} r_{ij}=1 & (i,j=1,2,3\ldots\ldots 9,10,11)\\ r_{ij}\times r_{ji}=0 & (i,j=2,3\ldots\ldots 9,10,11)\end{cases}$$

5.1.4 模型求解域结果分析：

在这里我们引入以下符号：

d_{ij}——第i个景点和第j个景点之间的路程；

v——旅客们所乘坐的旅游大巴的平均时速，v=50km/h；
m——旅客们所乘坐的旅游大巴的平均费用，h=0.25 元/h；

通过上网查询资料，我们可以得到d_{ij}的具体值，根据公式$t_{ij}=d_{ij}/v$可得到相应的t_{ij}，同样根据公式$c_{ij}=d_{ij}\times m$可以得到相应的c_{ij}（i，j=1，2，……，11）。（d_{ij}、t_{ij}和c_{ij}的具体数值见附录）

同样，通过对济南的一些旅行社进行咨询，我们得出旅客在第i个景点的最佳逗留时间和他们在第i个景点总消费：

t1	t2	t3	t4	t5	t6	t7	t8	t9	t10
36	24	5	4	24	8	18	10	6	32

（单位：小时）

c1	c2	c3	c4	c5	c6	c7	c8	c9	c10
240	180	130	50	400	170	320	90	148	450

（单位：元）

模型一的结果：

根据模型，使用 Lingo 编程[2]求出最短路线，综合实际条件做适当调整得出较优路线如下：

旅游景点数 n	10
每人总花费 c（单位：元）	1380
路线	趵突泉公园+泉城广场→大明湖→泉城公园→英雄山烈士陵园→洪家楼广场→黄河森林公园→金象山乐园→朱家峪→千佛山→济南动物园→趵突泉公园+泉城广场

5.2、 模型二的建立与求解

5.2.1 目标函数的确立

这里我们考虑另一种较好的费用确定：在一定的时间和费用内，游览最多的景点。因此，我们设计的路线必须满足时间和费用的要求，所以时间和费用便成了约束条件。我们考虑控制变量，将问题的约束条件分离出来。求解出目标函数。

(1)费用限制，最多景点，时间不限。（控制时间变量）

令决策变量为 $r_{ij}=\begin{cases}1 & \text{旅客从第i景点到第j个景点}\\ 0 & \text{其他}\end{cases}$

目标函数： $\min\sum_{i=1}^{n}\sum_{j=1}^{n}w_{ij}r_{ij}$

约束条件 $\sum_{i=1}r_{ij}=1,\sum_{j=1}r_{ij}=1$ （i,j=1，2，3……9, 10, 11）

x_{ij} 必须形成一条巡回路线

由于 Hamilton – 圈【3】问题属 $NP-$ 完全问题，且对于这 11 个点的图(还要增加 m – 1 个点）来说，要求得真正的最优解是不太现实的，于是我们考虑根据几何观采用启发式算法，来求得近似最优解。

首先要做的是在满足相应的条件下，先大致确定一下景点的个数。

旅游的总费用由两部分组成，分别为交通总费用和旅游景点的花费。我们的目标函数为：

$$\sum_{i=1}^{n}\sum_{j=1}^{n}c_{ij}r_{ij}+\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}r_{ij}\left(b_i+b_j\right)+\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}r_{ij}\left(d_i+d_j\right)\le 2000$$

①交通总花费

因为 c_{ij} 表示从第 i 个景点到第 j 个景点所需的交通费用，因此我们可以很容易的得到交通总费用为：

$$a_1=\sum_{i=1}^{n}\sum_{j=1}^{n}r_{ij}\times c_{ij}$$

②旅游景点的花费

b_i+d_i 表示该旅游者在 i 个景点的消费，r_{ij} 可以表示出旅客们是否到达第 i 个和第 j 个景点，而整个旅游路线又是一个环形，因此实际上将旅客在景点的花费计算了两遍，从而我们可得旅游景点的花费为：

$$b_i+d_i=\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}r_{ij}\left(b_i+b_j\right)+\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}r_{ij}\left(d_i+d_j\right)$$

③约束：0——1 变量约束

因为题目要求从泉城广场站出发，再回到泉城广场站，那么我们可以把所有的景点连成一个圈，而把每一个景点看做圈上一个点。对于每个点来说，只允许最多一条边进入，同样只允许最多一条边出来，并且只要有一条边进入就要有一条边出去，该题在前面已经得到证明，这里不再证明。所以约束为：

$$\sum_{i}^{n}r_{ij}\le\sum_{j}^{n}r_{ij}\le 1 \qquad （i,j=1，2，3\cdots\cdots 9,10,11）$$

当 $i=1$ 时，因为泉城广场站是出发点，所以 $\sum_{i=1}r_{ij}=1$；

$j=1$ 时，因为旅客最终要回到泉城广场，所以 $\sum_{j=1}r_{ij}=1$.

根据题目所述，我们可以得到以下结论：

$$\sum_{i}r_{ij}=\sum_{j}r_{ij}\le 1$$

$$\sum_{i=1}r_{ij}=1 \qquad \sum_{j=1}r_{ij}=1 \qquad （i,j=1，2，3\cdots\cdots 9,10,11）$$

同样，当i， $j \geq 2$时，根据题意不可能出现$r_{ij}=r_{ji}=1$，即不可能出现游客在两地间往返旅游，因为本题为一次巡回，则综上所述，我们可得约束为：

$$r_{ij} \times r_{ji} = 0$$

（2）费用不限，最多景点，时间限制 （控制费用变量）

设计一个家庭旅游套餐，满足需求--费用不限，最多景点，时间限制。

在实际问题中由于受到约束条件限制，如交通班次不衔接、到达目的地景点关闭、住宿等问题，导致无法从全局求解出最优路线，因此，我们选用贪婪法从局部最优求解此问。

显然，时间限制和游览的景点尽量多是该问题的两个目标。

$$MinT = m_1 + m_2 + m_3 = \sum_{i=1}^{11}\sum_{j=1}^{11} t_{ij} r_{ij} + + \frac{1}{2}\sum_{i=1}^{11}\sum_{j=1}^{11} r_{ij}\left(t_i + t_j\right) + \frac{1}{2}\sum_{i=1}^{11}\sum_{j=1}^{11} r_{ij}\left(e_i + e_j\right)$$

约束条件：

$$\begin{cases} \sum_{i=1}^{11}\sum_{j=1}^{11} r_{ij} = 11 & (i, j=1，2，3\cdots\cdots 9, 10, 11) \\ \sum_{i=1} r_{ij} = 1, \sum_{j=1} r_{ij} = 1 & (i, j=1，2，3\cdots\cdots 9, 10, 11) \\ r_{ij} \times r_{ji} = 0 & (i, j=2，3\cdots\cdots 9, 10, 11) \end{cases}$$

贪婪法[3]：

利用贪婪法从旅游耗时、路程长短、住宿耗时几方面综合考虑，确定局部最优解。首先，我们先删除耗时最多和相对距离较远的景点：金象山和朱家裕。剩下的景点相对比较聚集且景点逗留时间较短，我们每一次都从与本地相邻的几个城市中选择耗时最短并且满足交通班次与景点逗留时间不产生矛盾的目的地，达到局部优化时隙分配的目的。依次类推，确定一条包揽尽可能多景点的路线在5天之内返回泉城广场。

（3）费用限制，最多景点，时间限制

根据上面的分析，我们得到最终的目标函数为：

$$\sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij} r_{ij} + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} r_{ij}\left(b_i + b_j\right) + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} r_{ij}\left(d_i + d_j\right) \leq 2000\text{元}$$

$$\sum_{i=1}^{n}\sum_{j=1}^{n} t_{ij} r_{ij} + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} r_{ij}\left(t_i + t_j\right) + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} r_{ij}\left(e_i + e_j\right) \leq 120(h)$$

5.2.2 模型的求解与结果分析

根据模型，使用 Lingo 编程，得出结果如下：

旅游景点数 n	6

每人总花费 c（单位：元）	790
路线	趵突泉公园+泉城广场→大明湖→泉城公园→英雄山烈士陵园→洪家楼广场→趵突泉公园+泉城广场

5.3 模型三的建立与求解

5.3.1 目标函数的确立：

我们在本模型中所要实现的目标是，使旅客在 10 天时间内花最少的钱游览尽可能多的地方。显然，花费最少和游览的景点尽量多是该问题的两个目标。因此，我们的做法是在满足相应的约束条件下，先确定游览的景点数，然后计算出在这种情况下的最小花费。这样最终会得出几种旅游路线，而不同的家庭可以根据自己的实际情况进行选择。

游览的总费用由 3 部分组成，分别为交通总费用、在旅游景点的花费和旅客可能的花费。我们定义：

m——旅客的旅游总花费；

m_1——旅客的交通总费用；

m_2——旅客的旅游景点的花费；

m_3——旅客可能花费的费用（住宿费、吃饭的费用等）；

从而得到目标函数： Min $m = m_1 + m_2 + m_3$

（1）交通总花费

因为c_{ij}表示从第i个景点到第j个景点所需的交通费用，而r_{ij}是判断旅客们是否从第i个景点直接到第j个景点的 0—1 变量，因此我们可以很容易的得到交通总费用为：

$$m_1 = \sum_{i=1}^{11}\sum_{j=1}^{11} r_{ij} \times c_{ij}$$

（2）旅游景点的花费

因为c_i表示旅客们在i个景点的总消费，r_{ij}也可以表示出旅客们是否到达过第i个和第j个景点，而整个旅游路线又是一个环形，因此$\sum_{i=1}^{11}\sum_{j=1}^{11} r_{ij} \times \left(c_i + c_j\right)$实际上将旅客们在所到景点的花费计算了两遍，从而我们可得旅游景点的花费为：

$$m_2=\frac{1}{2}\times\sum_{i=1}^{11}\sum_{j=1}^{11}r_{ij}\times(c_i+c_j)$$

（3）可能的费用

用d_i表示旅客在第i个景点消费费用，其中有住宿费、吃饭费用和其他方面的费用；用r_{ij}表示旅客是否从第i个景点到达第j景点的0---1变量，由对本题的分析可知，本题为环形路线，且是一次巡回，所以我们得到可能的费用为：

$$m_3=\frac{1}{2}\times\sum_{i=1}^{11}\sum_{j=1}^{11}r_{ij}\times(f_i+f_j)$$

从而我们可以得到目标函数为：

$$\text{Min}\quad m=m_1+m_2+m_3$$

$$=\sum_{i=1}^{11}\sum_{j=1}^{11}r_{ij}\times c_{ij}+\frac{1}{2}\times\sum_{i=1}^{11}\sum_{j=1}^{11}r_{ij}\times(c_i+c_j)+\frac{1}{2}\times\sum_{i=1}^{11}\sum_{j=1}^{11}r_{ij}\times(f_i+f_j)$$

5.3.2 约束条件：

①时间约束

对于有时间限定的家庭，我们把他们在济南的旅游时间设置为不多于10天(供旅游时间为120小时)，而这些时间包括在路途中的时间和在旅游景点逗留的时间。因为t_{ij}表示从第i个景点到第j个景点路途中所需时间，所以路途中所需总时间为$\sum_{i=1}^{11}\sum_{j=1}^{11}r_{ij}\times t_{ij}$；$t_i$表示旅客们在第$i$个景点的逗留时间，故旅客们在旅游景点的总逗留时间为 $\frac{1}{2}\times\sum_{i=1}^{11}\sum_{j=1}^{11}r_{ij}\times(t_i+t_j)$。因此，总的时间约束为：

$$\sum_{i=1}^{11}\sum_{j=1}^{11}r_{ij}\times t_{ij}+\frac{1}{2}\times\sum_{i=1}^{11}\sum_{j=1}^{11}r_{ij}\times(t_i+t_j)\le 120$$

②旅游景点数约束

根据假设，整个旅游路线是环形，即最终旅客们要回到泉城广场，因此$\sum_{i=1}^{11}1\sum_{j=1}^{11}r_{ij}$即表示旅客们旅游的景点数，这里我们假定要旅游的景点数为n（n=2，3,……，11）。因此旅游景点数约束为：

$$\sum_{i=1}^{11}\sum_{j=1}^{11}r_{ij}=n\qquad（n=2，3,……，11）$$

③0——1 变量约束

我们可以把所有的景点连成一个圈，而把每一个景点看做圈上一个点。对于每个点来说，只允许最多一条边进入，同样只允许最多一条边出来，并且只要有一条边进入就要有一条边出去。因此可得约束：

$$\sum_i r_{ij} = \sum_j r_{ij} \le 1 \qquad （i，\ j=1，2,\cdots\cdots，11）$$

当$i=1$时，因为泉城广场是出发点，所以$\sum_{i=1} r_{ij} = 1$；

$j=1$时，因为旅客们最终要回到泉城广场，所以$\sum_{j=1} r_{ij} = 1$。

综合以上可知，

$$\sum_i r_{ij} = \sum_j r_{ij} \le 1 \qquad （i，\ j=1，2,\cdots\cdots，11）$$

$$\sum_{i=1} r_{ij} = 1 \qquad \sum_{j=1} r_{ij} = 1$$

同样，当i，$j \ge 2$时，根据题意不可能出现$r_{ij} = r_{ji} = 1$，即不可能出现游客在两地间往返旅游，因为这样显然不满足游览景点尽量多的原则。因此我们可得约束：

$$r_{ij} \times r_{ji} = 0 \qquad （i，\ j=2，3,\cdots\cdots，11）$$

5.3.3 **模型建立：**

综上所述，我们可以得到总的模型为：

$$\text{Min} \quad m = m_1 + m_2 + m_3$$

$$= \sum_{i=1}^{11}\sum_{j=1}^{11} r_{ij} \times c_{ij} + \frac{1}{2} \times \sum_{i=1}^{11}\sum_{j=1}^{11} r_{ij} \times \left(c_i + c_j\right) + \frac{1}{2} \times \sum_{i=1}^{11}\sum_{j=1}^{11} r_{ij} \times \left(f_i + f_j\right)$$

约束条件：

$$\begin{cases} \sum_{i=1}^{11}\sum_{j=1}^{11} r_{ij} \times t_{ij} + \frac{1}{2} \times \sum_{i=1}^{11}\sum_{j=1}^{11} r_{ij} \times \left(t_i + t_j\right) \le 120 \\ \sum_{i=1}^{11}\sum_{j=1}^{11} r_{ij} = n \qquad （n=2，3,\cdots\cdots，11） \\ \sum_i r_{ij} = \sum_j r_{ij} \le 1 \qquad （i，\ j=1，2,\cdots\cdots，11） \\ \sum_{i=1} r_{ij} = 1 \qquad \sum_{j=1} r_{ij} = 1 \end{cases}$$

$$r_{ij} \times r_{ji} = 0 \qquad （i，\ j=2，3,\cdots\cdots，11）$$

5.3.4 模型求解与结果分析：

根据模型，使用 Lingo 编程，得出结果如下表：

旅游景点数 n	2	3	4
每人总花费 m（单位：元）	200	340	560
路线	1→3→1	1→4→3→1	1→4→3→5→1

旅游景点数 n	5	6
每人总花费 m（单位：元）	670	740
路线	1→3→9→7→4→1	1→3→10→7→9→8→1

旅游景点数 n	7
每人总花费 m（单位：元）	890
路线	1→3→10→4→7→9→8→1

对于上述结果，我们的推荐为：

路线一：趵突泉公园+泉城广场→泉城公园→大明湖→黄河森林公园→趵突泉公园+泉城广场

旅游景点数：4　　人均费用：560 元；

路线二：趵突泉公园+泉城广场→大明湖→英雄山烈士陵园→朱家峪→泉城公园→趵突泉公园+泉城广场

旅游景点数：5　　人均费用：670 元；

路线三：趵突泉公园+泉城广场→大明湖→洪家楼广场→朱家峪→英雄山烈士陵园→金象山乐园→趵突泉公园+泉城广场

旅游景点数：6　　人均费用：860 元。

5.3.5 模型修改

在时间约束上，考虑到暑期的问题，可以放宽对时间的要求，以便顺利游览完所有景点。我们不妨假定限制的时间为两个月（720 个小时），同上一问可得：

$$\sum_{i=1}^{11}\sum_{j=1}^{11} r_{ij}\times t_{ij} + \frac{1}{2}\times\sum_{i=1}^{11}\sum_{j=1}^{11} r_{ij}\times\left(t_i + t_j\right)\leq 720$$

最终建立的模型不变，约束条件变为：

$$\begin{cases} \sum_{i=1}^{11}\sum_{j=1}^{11} r_{ij}\times t_{ij} + \frac{1}{2}\times\sum_{i=1}^{11}\sum_{j=1}^{11} r_{ij}\times\left(t_i + t_j\right) \le 720 \\ \sum_{i=1}^{11}\sum_{j=1}^{11} r_{ij} = 11 \qquad (i，\ j=1，2,\cdots\cdots，11) \\ \sum_{i} r_{ij} = \sum_{j} r_{ij} \le 1 \qquad (i，\ j=1，2,\cdots\cdots，11) \\ \sum_{i} r_{ij} = 1 \quad \sum_{j} r_{ij} = 1 \qquad (i，\ j=1，2,\cdots\cdots，11) \\ r_{ij}\times r_{ji} = 0 \qquad (i，\ j=2，3,\cdots\cdots，11) \end{cases}$$

改进后，根据模型，使用 Lingo 编程，得出结果为：

旅游景点数 n	8
每人总花费 m（单位：元）	920
路线	趵突泉公园+泉城广场→大明湖→洪家楼广场→济南动物园→朱家峪→黄河森林公园→英雄山烈士陵园→金象山乐园→趵突泉公园+泉城广场

5.4 模型四的建立与求解

5.4.1.1 问题的分析：

在设计旅游套餐时，需要考虑人数的因素。假设每组家庭有 5 名成员。若有两个家庭想要组团去参加，但是一个家庭由于某些原因必须要拖几天才能够去旅游。这时，由假设可知，参观景点的人数越多，每人承担的费用越少，因此为了减少旅行费用，我们应该尽量安排：两个家庭在同时旅游的几天内在同样的景点旅游。

5.4.1.2 数据的处理

类似上一问，我们定义：

$\lambda_i^{'}$——第 i 个旅游景点对于第一个家庭的权重；

$\lambda_i^{''}$——第 i 个旅游景点对于第二个家庭的权重。

5.4.1.3 目标函数的确立：

此问中，我们引入以下符号：

m ——旅游总花费；

$m_1{}'$——第一个家庭的交通总费用；

$m_1{}''$——第二个家庭的交通总费用；

$m_2{}'$——第一个家庭旅游景点的花费；

$m_2{}''$——第二个家庭旅游景点的花费。

m_3——两个家庭同时在一景点旅游比分别旅游节约的费用。

5.4.2、目标函数

由以上的假设和符号，我们可以很容易的得到总的目标函数为：

$$\text{Min} \quad m = m_1{}' + m_1{}'' + m_2{}' + m_2{}'' - m_3$$

而所得结果所对应的每个家庭的总花费为：

$$m = \sum_{i=1}^{11}\sum_{j=1}^{11} r_{ij} \times c_{ij} + \frac{1}{2} \times \frac{90}{100} \times \sum_{i=1}^{11}\sum_{j=1}^{11} r_{ij} \times \left(c_i + c_j\right)$$

定义：

$$r_{ij}{}' = \begin{cases} 1 & \text{第一个家庭直接从第}i\text{个景点到达第}j\text{个景点} \\ 0 & \text{其他} \end{cases}$$

$$r_{ij}{}'' = \begin{cases} 1 & \text{第二个家庭直接从第}i\text{个景点到达第}j\text{个景点} \\ 0 & \text{其他} \end{cases}$$

从而可以推得：

$$\begin{cases} m_1{}' = 5 \times \sum_{i=1}^{11}\sum_{j=1}^{11} \lambda_i{}' \times r_{ij}{}' \times c_{ij} \\ m_1{}'' = 5 \times \sum_{i=1}^{11}\sum_{j=1}^{11} \lambda_i{}'' \times r_{ij}{}'' \times c_{ij} \end{cases}$$

又因为假设参观景点的人数每增加一人，每个人在景点的费用就减少原价的1‰，因此可得：

$$\begin{cases} m_2{}' = \frac{1}{2} \times 5 \times 0.95 \times \sum_{i=1}^{11}\sum_{j=1}^{11} \lambda_i{}' \times r_{ij}{}' \times \left(c_i + c_j\right) \\ m_2{}'' = \frac{1}{2} \times 5 \times 0.95 \times \sum_{i=1}^{11}\sum_{j=1}^{11} \lambda_i{}'' \times r_{ij}{}'' \times \left(c_i + c_j\right) \end{cases}$$

5.4.3 节约的费用

定义：

$$\alpha_i=\begin{cases}1 & \text{两个家庭同时在第}i\text{个景点旅游}\\ 0 & \text{其他}\end{cases}$$

因为两组分别旅行时按照原价的 95％收费，而两组同时在同一景点旅游时按照原价的 90％收费，因此后者比前者便宜了定价的 5％，因此：

$$m_3=10\times 0.05\times\frac{1}{2}\times\sum_{j=1}^{11}\sum_{i=1}^{11}\gamma_{ij}\times\left(\alpha_i\times\lambda_i\times c_i+\alpha_j\times\lambda_j\times c_j\right)$$

5.4.4 约束条件

①时间约束

由题目可知，家庭是在暑假期间带领孩子旅行，因此在济南的旅游时间应该不多于 2 个月。由假设可知，除了人的基本作息时间，用在旅游的时间可以视为一天 12 个小时（总共 720 小时），而这些时间包括在路途中的时间和在旅游景点逗留的时间。因为t_{ij}表示从第i个景点到第j个景点路途中所需时间，所以两个家庭在路途中所需总时间分别为$\sum_{i=1}^{11}\sum_{j=1}^{11}r_{ij}\times t_{ij}$ 和$\sum_{i=1}^{11}\sum_{j=1}^{11}r^{''}{}_{ij}\times t_{ij}$ ；t_i表示旅客们在第i个景点的逗留时间，故两个家庭在旅游景点的总逗留时间分别为$\frac{1}{2}\times\sum_{i=1}^{11}\sum_{j=1}^{11}r_{ij}^{'}\times\left(t_i+t_j\right)$和$\frac{1}{2}\times\sum_{i=1}^{11}\sum_{j=1}^{11}r_{ij}^{''}\times\left(t_i+t_j\right)$。因此，总的时间约束为：

$$\sum_{i=1}^{11}\sum_{j=1}^{11}r_{ij}^{'}\times t_{ij}+\frac{1}{2}\times\sum_{i=1}^{11}\sum_{j=1}^{11}r_{ij}^{'}\times\left(t_i+t_j\right)\le 720$$

$$\sum_{i=1}^{11}\sum_{j=1}^{11}r_{ij}^{''}\times t_{ij}+\frac{1}{2}\times\sum_{i=1}^{11}\sum_{j=1}^{11}r_{ij}^{''}\times\left(t_i+t_j\right)\le 720$$

②旅游景点数约束

根据假设，整个旅游路线是环形，即最终两个家庭要回到泉城广场，因此$\sum_{i=1}^{11}\sum_{j=1}^{11}r_{ij}$即表示旅客们旅游的景点数，这里我们假定两个家庭要旅游的景点数均为n（n=2，3，……，11）。因此旅游景点数约束为：

$$\sum_{i=1}^{11}\sum_{j=1}^{11}r_{ij}^{'}=\sum_{i=1}^{11}\sum_{j=1}^{11}r_{ij}^{''}=n$$

③0——1 变量约束

我们可以把所有的景点连成一个圈，而把每一个景点看做圈上一个点。对于每个点来说，只允许最多一条边进入，同样只允许最多一条边出来，并且只要有一条边进入就要有一条边出去。因此可得约束：

$$\sum_{i} r_{ij}^{'} = \sum_{j} r_{ij}^{'} \le 1$$

$$\sum_{i} r_{ij}^{''} = \sum_{j} r_{ij}^{''} \le 1 \quad （i，\ j=2,\cdots\cdots，11）$$

当 $i=1$ 时，因为泉城广场是出发点，所以 $\sum_{i=1} r_{ij}^{'} = 1$ 并且 $\sum_{i=1} r_{ij}^{''} = 1$；

当 $j=1$ 时，因为旅客们最终要回到泉城广场，所以 $\sum_{j=1} r_{ij}^{'} = 1$ 并且 $\sum_{j=1} r_{ij}^{''} = 1$。

综合以上可知，

$$\sum_{i} r_{ij}^{'} = \sum_{j} r_{ij}^{'} \le 1$$

$$\sum_{i} r_{ij}^{''} = \sum_{j} r_{ij}^{''} \le 1 \qquad （i，\ j=2,\cdots\cdots，11）$$

$$\sum_{i=1} r_{ij}^{'} = 1 \qquad \sum_{i=1} r_{ij}^{''} = 1$$

$$\sum_{j=1} r_{ij}^{'} = 1 \qquad \sum_{j=1} r_{ij}^{''} = 1$$

同样，当 $i，\ j \ge 2$ 时，根据题意不可能出现 $r_{ij}^{'} = r_{ji}^{'} = 1$ 和 $r_{ij}^{''} = r_{ji}^{''} = 1$，即不可能出现游客在两地见往返旅游，因为这样显然不满足游览景点尽量多的原则。因此我们可得约束：

$$r_{ij}^{'} \times r_{ji}^{'} = 0$$

$$r_{ij}^{''} \times r_{ji}^{''} = 0 \qquad （i，\ j=2，3,\cdots\cdots，11）$$

5.4.5 模型建立：

综上所述，我们可以得到总的模型为：

$$\text{Min} \qquad m_{\lambda} = m_1^{'} + m_1^{''} + m_2^{'} + m_2^{''} - m_3$$

其中：

$$
\begin{cases}
m_1' = 5\times\sum_{i=1}^{11}\sum_{j=1}^{11}\lambda_i'\times r_{ij}'\times c_{ij} \\
m_1'' = 5\times\sum_{i=1}^{11}\sum_{j=1}^{11}\lambda_i''\times r_{ij}''\times c_{ij} \\
m_2' = \frac{1}{2}\times 5\times 0.95\times\sum_{i=1}^{11}\sum_{j=1}^{11}\lambda_i'\times r_{ij}'\times\left(c_i+c_j\right) \\
m_2'' = \frac{1}{2}\times 5\times 0.95\times\sum_{i=1}^{11}\sum_{j=1}^{11}\lambda_i''\times r_{ij}''\times\left(c_i+c_j\right) \\
m_3 = 10\times 0.05\times\frac{1}{2}\times\sum_{j=1}^{11}\sum_{i=1}^{11}\gamma_{ij}\times\left(\alpha_i\times\lambda_i\times c_i+\alpha_j\times\lambda_j\times c_j\right)
\end{cases}
$$

约束条件：

$$
\begin{cases}
\sum_{i=1}^{11}\sum_{j=1}^{11}r_{ij}'\times t_{ij}+\frac{1}{2}\times\sum_{i=1}^{11}\sum_{j=1}^{11}r_{ij}'\times\left(t_i+t_j\right)\le 720 \\
\sum_{i=1}^{11}\sum_{j=1}^{11}r_{ij}''\times t_{ij}+\frac{1}{2}\times\sum_{i=1}^{11}\sum_{j=1}^{11}r_{ij}''\times\left(t_i+t_j\right)\le 720 \\
\sum_{i=1}^{11}\sum_{j=1}^{11}r_{ij}' = \sum_{i=1}^{11}\sum_{j=1}^{11}r_{ij}'' = n \qquad (n=2,3,\cdots\cdots,11) \\
\sum_{i}r_{ij}' = \sum_{j}r_{ij}'\le 1 \qquad \sum_{i}r_{ij}'' = \sum_{j}r_{ij}''\le 1 \qquad (i,\ j=1,2,\cdots\cdots,11) \\
\sum_{i=1}r_{ij}' = 1 \qquad \sum_{i=1}r_{ij}'' = 1 \\
\sum_{j=1}r_{ij}' = 1 \qquad \sum_{j=1}r_{ij}'' = 1 \\
r_{ij}'\times r_{ji}' = 0 \qquad r_{ij}''\times r_{ji}'' = 0 \qquad (i,\ j=2,3,\cdots\cdots,11)
\end{cases}
$$

5.4.6 模型求解与结果分析：

使用 lingo 编程，得到最佳结果：

旅游景点数 n	5
每人总花费 c（单位：元）	750
路线	第一个家庭： 趵突泉公园+泉城广场→大明湖→千佛山→洪家楼广场→朱家裕→趵突泉公园+泉城广场

	第二个家庭： 　　　跨突泉公园+泉城广场→洪家楼广场→朱家裕→金象山乐园→趵突泉公园+泉城广场

即第一个家庭先行出发，在游览了大明湖和千佛山后前往洪家楼广场，与第二个家庭会合，两个家庭共同游览了洪家楼广场和金象山乐园，之后第一个家庭返回泉城广场，而第个家庭组则前往金象山游览。

5.5 模型五的建立与求解

5.5.1.1 模型的分析

本问在上一个模型的基础上考虑了天气的因素，相应的也就增加了一个目标即：使因阴雨天气而带来的旅游损失降到最低。对于旅游损失，我们定义为旅客们在景点逗留时所对应的阴雨天气概率的总和。

5.5.1.2 数据处理

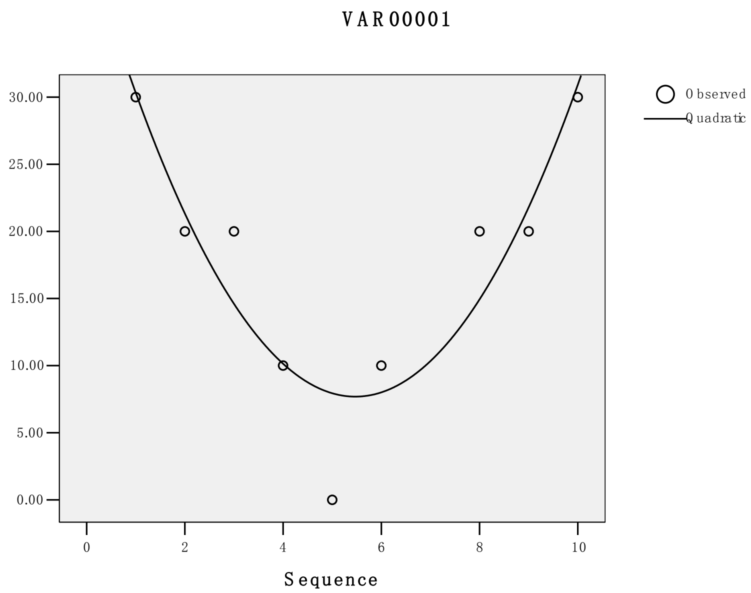
（1）通过查阅山东省济南市当地天气网得到附件二，并对数据进行处理

Ⅰ. 对于附件中超过 100％的数据我们修定其为 100％；

Ⅱ. 对于附件中缺失的数据，我们使用 SPSS 软件[5]进行时间序列预测如下：

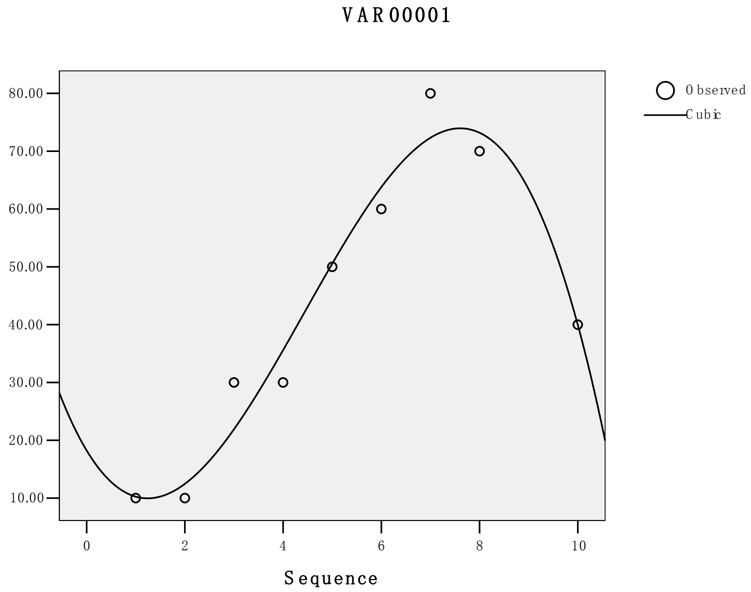
对于金象山的降水概率，最优拟合曲线为二次曲线，拟合结果为：金象山第七天降雨的概率为 10.33898％，我们取 10％。

拟合曲线图如下：



对于朱家裕的降水概率，最优拟合曲线为三次曲线，拟合结果为：朱家裕第九天降水的概率为 63.39119％，我们取 63％.

拟合曲线图如下：



综上我们得到最终的矩阵：$\left[P_{is}\right]_{11\times5}$（见附录）。

（2）数据的归一化处理

通过观察数据，我们发现旅游总花费和阴雨天气带来的旅游损失的数值差距较大，在利用二者综合确立目标时，为了避免其的影响，采用数据常用处理方法——极差变化法[6]，将数据做归一化处理。即：

$$C=\frac{c-c(n)_{\min}}{c(n)_{\max}-c(n)_{\min}} \quad ; \quad L=\frac{l-l(n)_{\min}}{l(n)_{\max}-l(n)_{\min}}$$

（3）确定目标函数

沿用上几个模型的思想，我们的做法是在满足相应的约束条件下，先确定游览的景点数，然后分别表示出相应的旅游总费用和阴雨天气带来的旅游损失，归一化处理后加权求最小值。这样最终会得出几种最佳方案，而旅客可以根据自己的实际情况进行选择。由此得到最终的目标函数：

$$\text{Min} \quad \mathrm{Q}=\gamma_1\times C+\gamma_2\times L$$

（其中 C,L 如上所述，$\gamma_1\ \gamma_2$ 为权重且 $\gamma_1+\gamma_2=1$）

Ⅰ.对于 C：

由上一个模型可知：$c = m_1'+m_1''+m_2'+m_2'' - m_3$，而相应的 $c(n)_{\min}$ 与 $c(n)_{\max}$ 也可以根据前几问的模型计算出（具体值见附录表格），因此我们用可以已知的结果来表达 C。

Ⅱ.对于 L：

在 L 的表达式中，关键是表示出 l，表示出 l 后 $l(n)_{\min}$ 和 $l(n)_{\max}$ 可以根据相应的模型通过编程很容易的计算出来。因为 l 表示阴雨天气带来的旅游损失，而我

们定义该损失为旅客们在景点逗留时所对应的阴雨天气概率的总和。这里我们设：

P_{is}——第i个景点在第s天阴雨的概率；（s=1，2，……10）

T_i——旅客们到达第i个景点的时间；

V——旅客们游览的景点的集合，如$V=\{1，8，5，7\}$表示旅客们游览了第 1、8、5、7 个景点。

因此旅客们会从第$[T_i]$天到第$[T_i+t_i]$天 都留在第i个景点（$[\ ]$表示取整）。综上，l

的可以表示为$\sum_{i\in V}\sum_{s=[T_I]}^{[T_I+t_i]}P_{is}$，这样我们也就可以表达出$L$。

5.5.2 约束条件

①时间约束

由题目可知，旅客们在川的旅游时间应该不多于两个月，而这些时间包括在路途中的时间和在旅游景点逗留的时间。因为t_{ij}表示从第i个景点到第j个景点路途中所需时间，所以路途中所需总时间为$\sum_{i=1}^{10}\sum_{j=1}^{10}r_{ij}\times t_{ij}$； t_i表示旅客们在第i个景点的逗留时间，故旅客们在旅游景点的总逗留时间为$\frac{1}{2}\times\sum_{i=1}^{10}\sum_{j=1}^{10}r_{ij}\times(t_i+t_j)$。因此，总的时间约束为：

$$\sum_{i=1}^{10}\sum_{j=1}^{10}r_{ij}\times t_{ij}+\frac{1}{2}\times\sum_{i=1}^{10}\sum_{j=1}^{10}r_{ij}\times(t_i+t_j)\le 720$$

②旅游景点数约束

根据假设，整个旅游路线是环形，即最终旅客们要回到泉城广场，因此$\sum_{i=1}^{10}\sum_{j=1}^{10}r_{ij}$即表示旅客们旅游的景点数，这里我们假定要旅游的景点数为n（n=2，3，……，10）。因此旅游景点数约束为：

$$\sum_{i=1}^{10}\sum_{j=1}^{10}r_{ij}=n \quad （n=2，3，……，10）$$

③0——1 变量约束

我们可以把所有的景点连成一个圈，而把每一个景点看做圈上一个点。对于每个点来说，只允许最多一条边进入，同样只允许最多一条边出来，并且只要有一条边进入就要有一条边出去。因此可得约束：

$$\sum_i r_{ij} = \sum_j r_{ij} \le 1 \qquad (i, \ j=1, 2, \cdots\cdots, 10)$$

当$i=1$时，因为泉城广场是出发点，所以$\sum_{i=1} r_{ij} = 1$；

当$j=1$时，因为旅客们最终要回到泉城广场，所以$\sum_{j=1} r_{ij} = 1$。

综合以上可知，

$$\sum_i r_{ij} = \sum_j r_{ij} \le 1 \qquad (i, \ j=1, 2, \cdots\cdots, 10)$$

$$\sum_{i=1} r_{ij} = 1 \qquad \sum_{j=1} r_{ij} = 1$$

同样，当i，$j \ge 2$时，根据题意不可能出现$r_{ij} = r_{ji} = 1$，即不可能出现游客在两地间往返旅游，因为这样显然不满足游览景点尽量多的原则。因此我们可得约束：

$$r_{ij} \times r_{ji} = 0 \qquad (i, \ j=2, 3, \cdots\cdots, 10)$$

④对T_i的约束

因为T_i表示旅客们到达第i个景点的时间，因此T_i应该等于到达第$i-1$个景点的时间与第$i-1$个景点到第i个景点途中时间的和。从而可得：

$$T_i = T_{i-1} + t_{i-1,i} \qquad (i=2, 3, \cdots\cdots, 10)$$

5.5.3 模型建立：

综上所述，我们可以得到总的模型为：

$$\text{Min} \quad Q = \gamma_1 \times C + \gamma_2 \times L$$

约束条件：

$$\begin{cases} \sum_{i=1}^{10}\sum_{j=1}^{10} r_{ij} \times t_{ij} + \frac{1}{2} \times \sum_{i=1}^{10}\sum_{j=1}^{10} r_{ij} \times (t_i + t_j) \le 720 \\ \sum_{i=1}^{10}\sum_{j=1}^{10} r_{ij} = k \\ \sum_i r_{ij} = \sum_j r_{ij} \le 1 \qquad (i, \ j=1, 2, \cdots\cdots, 10) \\ r_{ij} \times r_{ji} = 0 \qquad (i, \ j \ge 2) \\ T_i = T_{i-1} + t_{i-1,i} \qquad (i=2, 3, \cdots\cdots, 10) \end{cases}$$

5.5.4 模型结果

由于结果数据庞大，我们在这里只列出$n=5$的情形。

当$\gamma_1=0.6,\gamma_2=0.4$时

人均总费用 C（单位：元）	损失
660	1.7
路线：1→10→3→4→8→1	

当$\gamma_1=0.4,\gamma_2=0.6$时

人均总费用 C（单位：元）	损失
730	1.6
路线：1→7→9→8→4→1	

当$\gamma_1=0.7,\gamma_2=0.3$时

人均总费用 C（单位：元）	损失
640	1.8
路线：1→10→3→7→4→1	

当$\gamma_1=0.3,\gamma_2=0.7$时

人均总费用 C（单位：元）	损失
780	0.9
路线：1→10→9→8→4→1	

当$\gamma_1=0.5,\gamma_2=0.5$时

人均总费用 C（单位：元）	损失
680	1.6
路线：1→10→9→8→4→1	

由以上结果，我们推荐路线为：趵突泉公园+泉城广场→洪家楼广场→英雄山烈士陵园→金象山乐园→泉城公园→趵突泉公园+泉城广场，相应人均消费9680元，阴雨天气带来的损失为1.6。

六、模型的评价与推广

6.1、模型的优缺点

模型的优点：

1.本文思路清晰，模型恰当，得出的方案合理；

2.本文成功的使用了0—1变量，使模型的建立和编程得以顺利进行；

3.我们在程序中采用了TCP算法，简化了模型的求解难度；

4.阴雨天气：由于数据庞大，对程序的要求很高，尽管经过了检验，但结果依然比较粗糙，有待进行进一步的改进。

5.将路径某某某问题转化为最优Hamilton路问题。对本问题，采用贪婪算法，得到的结果不仅非常接近理论结果，而且计算复杂度仅为$O\left(n^2\right)$。

模型的缺点：

因数据资料搜集的不完整，准确性也有待商榷，而且没有对最终方案进行更为细致的讨论研究，这些方面有待改进。

6.2 模型的改进与推广

1.实际情况中，两景点之间可能还有出公路外其他交通方式，如航班、铁路，增加这些考虑后，结果会更加合理。

2.在解决时间有限，费用有限，最多景点的问题时，我们可以进行题目优化，将问题优化为费用有限，总满意度尽可能大。之所以做这样的优化，是因为考虑到旅游是一种放松型娱乐活动，在花钱尽量少的前提下，能让游客产生满意度的因素不仅仅是游览地点的多少，还需考虑如下几个因素：

(1)不同的旅游景点给人带来的满意程度是不一样的;

(2)旅游难免给人带来一定的消极感受，主要是坐车时间的长短以及行走线路的平坦度。

因此尽量减少这种消极因素对人的影响。

(3)从心理学角度，如果一个游客非常向往去某个景点旅游而却没有去成的话，他必然会产生一定的遗憾度。因此，希望设计出来的旅游路线能给游客带来尽量大的享受和尽量小的遗憾度。

3.在建立模型的时候，我们可以考虑景区的参观时间以及景区分类，在设计家庭旅游套餐时可以考虑是否必须包括某种类型的景区以及相应类型景区的数目。

七、参考文献

[1]谢金星 薛毅,《优化建模与LINDO/LINGO软件》,北京:清华大学出版社,2005。

[2]百度百科，哈密顿回路，
http://baike.baidu.com/link?url=B9DfA2xywD4eNgA9ILlyanXLgp
qf1Kd88A0H1Bx7jud5D3Prv9EFTDCAqSidraxxdyrCeYUh4tDfB07-RgJf1K,
2014/5/25

[3]郑宗汉 郑晓明,《算法设计与分析》, 北京：清华大学出版社，2005

[4]薛薇,《统计分析与SPSS的应用》, 北京：中国人民大学出版社，2008

[5]李美娟 陈国宏 陈衍泰,《综合评价中指标标准化方法研究》,《中国管理科学》, 第12卷 专辑：46-47,2004年10月

[6]杜端甫. 运筹图论(图、网络理论中的运筹问题)[M]. 北京航空航天大学出版社, 240- 258.

八、附录

程序及运行结果（由于数据庞大，只选择了部分数据）

（程序）

```
sets:
jingdian/1..10/:c,t,l;
!其中：1,2,...,10 分别代表趵突泉公园+泉城广场、千佛山、大明湖、泉城公园
```

```
、黄河森林公园、济南动物园、朱家峪、金象山乐园、英雄山烈士陵园、洪家楼广场；c，t 分别表示旅行团在各景点的吃住消费和逗留时间；w 表示各景点选择性权重；l 是为了控制不出现两个以上环形回路而设的一个变量;
links(jingdian,jingdian):r,cc,tt;
!其中：r 为 0-1 变量（0 表示两景点不相连，1 表示两景点相通）；cc 为两景点之间的交通费用；tt 为两景点之间的交通时间;
endsets
data:
t=7 24 18 12 36 30 12 9 15 24 17;
c=120 423 300 135 378 390 175 90 148 303 241;
tt=0	8.54	4.74	2.82	3.44	5.08	8.4	1.32	1.54	6.14	6.6
8.54	0	1.22	11.52	12.14	10.9	13.1	8.84	8.98	14.84	15.54
4.74	1.22	0	11.22	11.82	9.38	11.58	7.66	7.46	13.44	13.9
2.82	11.52	11.22	0	0.88	7.78	8.08	4.02	4.24	5.84	6.3
3.44	12.14	11.82	0.88	0	8.42	8.24	4.66	4.88	6	6.46
5.08	10.9	9.38	7.78	8.42	0	2.18	4.24	4.04	5.98	6.74
8.4	13.1	11.58	8.08	8.24	2.18	0	6.08	6.22	3.86	2.86
1.32	8.84	7.66	4.02	4.66	4.24	6.08	0	0.3	6.28	6.74
1.54	8.98	7.46	4.24	4.88	4.04	6.22	0.3	0	6.08	6.54
6.14	14.84	13.44	5.84	6	5.98	3.86	6.28	6.08	0	2.08
6.6	15.54	13.9	6.3	6.46	6.74	2.86	6.74	6.54	2.08	0;
!其中：主对角线为零,表示各景点到自身交通费用为零;
cc=0	128.1	71.1	42.3	51.6	76.2	126	19.8	23.1	92.1	99
128.1	0	18.3	172.8	182.1	163.5	196.5	132.6	134.7	222.6	233.1
71.1	18.3	0	168.3	177.3	140.7	173.7	114.9	111.9	201.6	208.5
42.3	172.8	168.3	0	13.2	116.7	121.2	60.3	63.6	87.6	94.5
51.6	182.1	177.3	13.2	0	126.3	123.6	69.9	73.2	90	96.9
76.2	163.5	140.7	116.7	126.3	0	32.7	63.6	60.6	89.7	101.1
126	196.5	173.7	121.2	123.6	32.7	0	91.2	93.3	57.9	42.9
19.8	132.6	114.9	60.3	69.9	63.6	91.2	0	4.5	94.2	101.1
23.1	134.7	111.9	63.6	73.2	60.6	93.3	4.5	0	91.2	98.1
92.1	222.6	201.6	87.6	90	89.7	57.9	94.2	91.2	0	31.2
99	233.1	208.5	94.5	96.9	101.1	42.9	101.1	98.1	31.2	0;
!其中：主对角线为零,表示各景点到自身的交通时间为零;
n=?;
!其中：n 表示计划游玩的景点数目;
enddata
```

```
min=@sum(jingdian(j):@sum(jingdian(i):r(i,j)*(cc(i,j)+0.5*(c(i)+c(j)))));
!目标函数：表示计划游玩的景点数目为n时的最小费用;
@for(jingdian(i):r(i,i)=0);
!约束条件：表示各景点到自身没有路线相连的约束条件;
@for(jingdian(i)|i#ge#2:@for(jingdian(j)|j#ge#2:r(i,j)+r(j,i)<1));
!约束条件：表示除起点（泉城广场）外，若旅行团从景点i到景点j去游玩（即r(i,j)=1），则不会再从景点j到景点i去游玩（即r(j,i)=0），也就是说除起点外每个景点只游玩一次;
a=@sum(jingdian(j):@sum(jingdian(i):r(i,j)*(tt(i,j)+0.5*(t(i)+t(j)))));
@sum(jingdian(j):@sum(jingdian(i):r(i,j)*(tt(i,j)+0.5*(t(i)+t(j)))))<120;
!约束条件:表示总的旅行时间（交通时间和景点逗留时间）不超过给定时间10天120小时;
@for(jingdian(i):@sum(jingdian(j):r(i,j))=@sum(jingdian(j):r(j,i)));
@for(jingdian(i)|i#eq#1:@sum(jingdian(j):r(i,j))=1);
@for(jingdian(i)|i#ne#1:@sum(jingdian(j):r(i,j))<1);
!这三个约束条件：表示起点有且仅有一条路线出去和一条路线进来，其它景点要么有且仅有一条路线出去和一条路线进来，要么既没有路线出去也没有路线进来;
@for(links:@bin(r));
!约束条件:表示0-1变量约束;
@sum(jingdian(j):@sum(jingdian(i):r(i,j)))=n;
!约束条件:表示旅游景点的数目为n的约束;
@for(jingdian(i):@for(jingdian(j)|j#gt#1#and#j#ne#i:l(j)>=l(i)+r(i,j)-(n-2)*(1-r(i,j))+(n-3)*r(j,i)));
@for(jingdian(i)|i#gt#1:l(i)<n-1-(n-2)*r(1,i);l(i)>1+(n-2)*r(i,1));
!这两个约束条件：为了控制不出现两个以上环形回路，保证有且仅有一条环形路线;
```

结果：（以 *n*=5 为例）

由于数据庞大，只剪切出重要的部分如下：

Global optimal solution found at iteration: 2042

Objective value: 670

Variable	Value	Reduced Cost
N	5.000000	0.000000
R(1, 4)	1.000000	120
R(4, 7)	1.000000	130
R(7, 9)	1.000000	170
R(8, 1)	1.000000	110
R(9, 8)	1.000000	140

```
sets:
jingdian/1..10/:c,t,l;
links(jingdian,jingdian):r,cc,tt;
!其中：r为0-1变量（0表示两景点不相连，1表示两景点相通）；cc为两景点之
```

```
间的交通费用；tt 为两景点之间的交通时间;
endsets
data:
t=7 24 18 12 36 30 12 9 15 24 17;
c=120 423 300 135 378 390 175 90 148 303 241;
tt=0 8.54	4.74	2.82	3.44	5.08	8.4	1.32	1.54	6.14	6.6
8.54	0	1.22	11.52	12.14	10.9	13.1	8.84	8.98	14.84
	15.54
4.74	1.22	0	11.22	11.82	9.38	11.58	7.66	7.46	13.44
	13.9
2.82	11.52	11.22	0	0.88	7.78	8.08	4.02	4.24	5.84	6.3
3.44	12.14	11.82	0.88	0	8.42	8.24	4.66	4.88	6	6.46
5.08	10.9	9.38	7.78	8.42	0	2.18	4.24	4.04	5.98
	6.74
8.4	13.1	11.58	8.08	8.24	2.18	0	6.08	6.22	3.86	2.86
1.32	8.84	7.66	4.02	4.66	4.24	6.08	0	0.3	6.28	6.74
1.54	8.98	7.46	4.24	4.88	4.04	6.22	0.3	0	6.08	6.54
6.14	14.84	13.44	5.84	6	5.98	3.86	6.28	6.08	0	2.08
6.6	15.54	13.9	6.3	6.46	6.74	2.86	6.74	6.54	2.08	0;
!其中：主对角线为零,表示各景点到自身交通费用为零;
cc=0	128	71	42	52	77	126	20	23	92	99
128	0	18	173	182	164	197	133	135	223	233
71	18	0	168	177	141	174	115	112	202	209
42	173	168	0	13	117	121	60.	64	88	95
52	182	177	13	0	126	124	70	73	90	97
76	164	141	117	126	0	33	64	61	90	101
126	197	174	121	124	33	0	91	93	58	43
20	133	115	60	70	64	91	0	5	94	101
23	135	112	64	73	61	93	5	0	91	98
92	223	202	88	90	90	58	94	91	0	31
99	233	209	95	97	101	43	101	98	31	0;
!其中：主对角线为零,表示各景点到自身的交通时间为零;
enddata
min=@sum(jingdian(j):@sum(jingdian(i):r(i,j)*(cc(i,j)+0.5*(c(i)+c(j)))));
!目标函数：表示计划游玩的景点数目为 n 时的最小费用;
@for(jingdian(i):r(i,i)=0);
!约束条件：表示各景点到自身没有路线相连的约束条件;
@for(jingdian(i)|i#ge#2:@for(jingdian(j)|j#ge#2:r(i,j)+r(j,i)<1));
!约束条件：表示除起点外，若旅行团从景点 i 到景点 j 去游玩（即 r(i,j)=1），则不会再从景点 j 到景点 i 去游玩（即 r(j,i)=0），也就是说除起点外每个景点只游玩一次;
a=@sum(jingdian(j):@sum(jingdian(i):r(i,j)*(tt(i,j)+0.5*(t(i)+t(j)))));
@sum(jingdian(j):@sum(jingdian(i):r(i,j)*(tt(i,j)+0.5*(t(i)+t(j)))))<360;
!约束条件:表示总的旅行时间（交通时间和景点逗留时间）不超过给定时间 30
```

天 360 小时;
@for(jingdian(i):@sum(jingdian(j):r(i,j))=@sum(jingdian(j):r(j,i)));
@for(jingdian(i)|i#eq#1:@sum(jingdian(j):r(i,j))=1);
@for(jingdian(i)|i#ne#1:@sum(jingdian(j):r(i,j))<1);
!这三个约束条件：表示起点有且仅有一条路线出去和一条路线进来，其它景点要么有且仅有一条路线出去和一条路线进来，要么既没有路线出去也没有路线进来;
@for(links:@bin(r));
!约束条件:表示 0-1 变量约束;
@sum(jingdian(j):@sum(jingdian(i):r(i,j)))=11;
!约束条件:表示旅游景点的数目为 n 的约束;
@for(jingdian(i):@for(jingdian(j)|j#gt#1#and#j#ne#i:l(j)>=l(i)+r(i,j)-(n-2)*(1-r(i,j))+(n-3)*r(j,i)));
@for(jingdian(i)|i#gt#1:l(i)<n-1-(n-2)*r(1,i);l(i)>1+(n-2)*r(i,1));
!这两个约束条件：为了控制不出现两个以上环形回路，保证有且仅有一条环形路线；
结果（以 n=5 为例）：
Local optimal solution found at iteration: 390
Objective value: 920

Variable	Value	Reduced Cost
R(1, 4)	1.000000	0.000000
R(2, 3)	1.000000	0.000000
R(3, 1)	1.000000	0.000000
R(4, 5)	1.000000	0.000000
R(5, 10)	1.000000	0.000000
R(6, 9)	1.000000	0.000000
R(7, 6)	1.000000	0.000000
R(8, 2)	1.000000	0.000000
R(9, 8)	1.000000	0.000000
R(10, 11)	1.000000	0.000000
R(11, 7)	1.000000	0.000000

sets:
jingdian/1..11/:c,rrv,t,w,w2,l,l2,v,v2,b,b2,y,tv,tv2,sv;!其中：gailv/1..10/;
links(jingdian,jingdian):r,r2,cc,tt,ssv,rrr,u1,u2,u3,u4,u5,u6,u7,u8,u9,u10,u11;!其中：r为0-1变量（0表示两景点不相连，1表示两景点相通）；cc为两景点之间的交通费用；tt为两景点之间的交通时间；
links2(jingdian,gailv):pp,pv;
links1(jingdian,jingdian,jingdian):x,x2,bb,bb2,ppv;
endsets
data:
t=7 24 18 12 36 30 12 9 15 24 17;!其中:t(1)=0,表示在泉城广场的逗留时间为0，

因为相对旅行团而言，由于泉城广场既是起点又是终点，并未在泉城广场游玩；
c=120 423 300 135 378 390 175 90 148 303 241;!其中：c(1)=0，表示在泉城广场的吃住费用算做，理由同上；
w=0 1 1 1.058 1.058 0.942 0.942 1.058 1.058 0.942 0.942;
w2=0 0.843 0.843 1.104 1.104 1.024 1.024 1.004 1.004 1.024 1.024;
cc=0 128 71 42 52 77 126 20 23 92 99
128 0 18 173 182 164 197 133 135 223 233
71 18 0 168 177 141 174 115 112 202 209
42 173 168 0 13 117 121 60. 64 88 95
52 182 177 13 0 126 124 70 73 90 97
76 164 141 117 126 0 33 64 61 90 101
126 197 174 121 124 33 0 91 93 58 43
20 133 115 60 70 64 91 0 5 94 101
23 135 112 64 73 61 93 5 0 91 98
92 223 202 88 90 90 58 94 91 0 31
99 233 209 95 97 101 43 101 98 31 0;!其中：主对角线为零,表示各景点到自身交通费用为零；
tt=0 8.54 4.74 2.82 3.44 5.08 8.4 1.32 1.54 6.14 6.6
8.54 0 1.22 11.52 12.14 10.9 13.1 8.84 8.98 14.84 15.54
4.74 1.22 0 11.22 11.82 9.38 11.58 7.66 7.46 13.44 13.9
2.82 11.52 11.22 0 0.88 7.78 8.08 4.02 4.24 5.84 6.3
3.44 12.14 11.82 0.88 0 8.42 8.24 4.66 4.88 6 6.46
5.08 10.9 9.38 7.78 8.42 0 2.18 4.24 4.04 5.98 6.74
8.4 13.1 11.58 8.08 8.24 2.18 0 6.08 6.22 3.86 2.86
1.32 8.84 7.66 4.02 4.66 4.24 6.08 0 0.3 6.28 6.74
1.54 8.98 7.46 4.24 4.88 4.04 6.22 0.3 0 6.08 6.54
6.14 14.84 13.44 5.84 6 5.98 3.86 6.28 6.08 0 2.08
6.6 15.54 13.9 6.3 6.46 6.74 2.86 6.74 6.54 2.08 0;!其中：主对角线为零,表示各景点到自身的交通时间为零；
z=0.95;
n2=6;
n=6;!其中：n表示计划游玩的景点数目；
enddata
min=@sum(jingdian(j):@sum(jingdian(i):w(j)*r(i,j)*(cc(i,j)+0.95*c(j)*(1-rrv(j)*(1-z)))))+@sum(jingdian(j):@sum(jingdian(i):w2(j)*r2(i,j)*(cc(i,j)+0.95*c(j)*(1-rrv(j)*(1-z)))));!目标函数：表示计划游玩的景点数目为n时的最小费用；w(j)为各景点权重的倒数或1-该权重所得；
feiyong=@sum(jingdian(j):@sum(jingdian(i):r(i,j)*(cc(i,j)+0.95*c(j)*(1-rrv(j)*(1-z)))));
feiyong2=@sum(jingdian(j):@sum(jingdian(i):r2(i,j)*(cc(i,j)+0.95*c(j)*(1-rrv(j)*(1-z)))));
@sum(jingdian(j):@sum(jingdian(i):w(j)*r(i,j)*(cc(i,j)+0.95*c(j)*(1-rrv(j)*(1-z)))))+
@sum(jingdian(j):@sum(jingdian(i):w2(j)*r2(i,j)*(cc(i,j)+0.95*c(j)*(1-rrv(j)*(1-z)))))
>1493;

```
@for(jingdian(i):r(i,i)=0);!约束条件：表示各景点到自身没有路线相连的约束条件;
@for(jingdian(i):r2(i,i)=0);
@for(jingdian(i)|i#ge#2:@for(jingdian(j)|j#ge#2:r(i,j)+r(j,i)<1));!约束条件：表示除起点外，若旅行团从景点i到景点j去游玩（即r(i,j)=1），则不会再从景点j到景点i去游玩（即r(j,i)=0），也就是说除起点外每个景点只游玩一次;
@for(jingdian(i)|i#ge#2:@for(jingdian(j)|j#ge#2:r2(i,j)+r2(j,i)<1));
@sum(jingdian(j):@sum(jingdian(i):r(i,j)*(tt(i,j)+t(j))))<120;!约束条件:表示总的旅行时间（交通时间和景点逗留时间）不超过给定时间10天;
@sum(jingdian(j):@sum(jingdian(i):r2(i,j)*(tt(i,j)+t(j))))<120;
@for(jingdian(i):@sum(jingdian(j):r(i,j))=@sum(jingdian(j):r(j,i)));
@for(jingdian(i):@sum(jingdian(j):r2(i,j))=@sum(jingdian(j):r2(j,i)));
@for(jingdian(i)|i#eq#1:@sum(jingdian(j):r(i,j))=1);
@for(jingdian(i)|i#eq#1:@sum(jingdian(j):r2(i,j))=1);
@for(jingdian(i)|i#ne#1:@sum(jingdian(j):r(i,j))<1);!这三个约束条件:表示起点(成都)有且仅有一条路线出去和一条路线进来，其它景点要么有且仅有一条路线出去和一条路线进来，要么既没有路线出去也没有路线进来;
@for(jingdian(i)|i#ne#1:@sum(jingdian(j):r2(i,j))<1);
@for(links:@bin(r));!约束条件:表示0-1变量约束;
@for(links:@bin(r2));
@for(jingdian:@bin(rrv));
@for(links:@bin(rrr));
@sum(jingdian(j):@sum(jingdian(i):r(i,j)))=n;
@sum(jingdian(j):@sum(jingdian(i):r2(i,j)))=n2;
!@sum(jingdian(j):@sum(jingdian(i):r(i,j)))>n-0.5;
!@sum(jingdian(j):@sum(jingdian(i):r(i,j)))<n+0.5;
!@sum(jingdian(j):@sum(jingdian(i):r2(i,j)))<n2+0.5;!约束条件:表示旅游景点的数目为n的约束;
!@sum(jingdian(j):@sum(jingdian(i):r2(i,j)))>n2-0.5;
@for(jingdian(i):@for(jingdian(j)|j#gt#1#and#j#ne#i:l(j)>=l(i)+r(i,j)-(n-2)*(1-r(i,j))+(n-3)*r(j,i)));
@for(jingdian(i):@for(jingdian(j)|j#gt#1#and#j#ne#i:l2(j)>=l2(i)+r2(i,j)-(n2-2)*(1-r2(i,j))+(n2-3)*r2(j,i)));
@for(jingdian(i)|i#gt#1:l(i)<n-1-(n-2)*r(1,i);l(i)>1+(n-2)*r(i,1));!这两个约束条件：为了控制不出现两个以上环形回路，保证有且仅有一条环形路线;
@for(jingdian(i)|i#gt#1:l2(i)<n2-1-(n2-2)*r2(1,i);l2(i)>1+(n2-2)*r2(i,1));
@for(jingdian(i)|1#eq#i:v(i)=1;b(i)=0);
!@for(jingdian(k)|1#lt#k#and#k#le#n:@for(jingdian(i):@for(jingdian(j):x(k,i,j)=@if(r(i,j)#eq#1#and#v(k-1)#eq#i,j,0)));!v(k)=@sum(jingdian(j):@sum(jingdian(i):x(k,i,j))));
@for(jingdian(k)|1#lt#k#and#k#le#n:@for(jingdian(i):@for(jingdian(j):x(k,i,j)=@if(0.5#le#r(i,j)#and#r(i,j)#le#1.5#and#(i-0.5)#le#v(k-1)#and#v(k-1)#le#(i+0.5),j,0)));v(k)=@sum(jingdian(j):@sum(jingdian(i):x(k,i,j))));
@for(jingdian(i)|1#eq#i:v2(i)=1;b2(i)=0);
```

```
!@for(jingdian(k)|1#lt#k#and#k#le#n2:@for(jingdian(i):@for(jingdian(j):x2(k,i,j)=@if(r(i,j)#eq#1#and#v2(k-1)#eq#i,j,0)));!v2(k)=@sum(jingdian(j):@sum(jingdian(i):x2(k,i,j))));
@for(jingdian(k)|1#lt#k#and#k#le#n2:@for(jingdian(i):@for(jingdian(j):x2(k,i,j)=@if(0.5#le#r2(i,j)#and#r2(i,j)#le#1.5#and#(i-0.5)#le#v2(k-1)#and#v2(k-1)#le#(i+0.5),j,0)));v2(k)=@sum(jingdian(j):@sum(jingdian(i):x2(k,i,j))));
@for(jingdian(i)|2#le#i#and#i#le#(n):@for(jingdian(k):@for(jingdian(j):bb(k,i,j)=@if(v(i)#eq#j#and#v(i-1)#eq#k,t(k)+tt(k,j),0)));b(i)=@sum(jingdian(j):@sum(jingdian(k):bb(k,i,j))));
@for(jingdian(k)|k#le#n:tv(k)=@sum(jingdian(i)|i#le#(k):b(i))/12);
@for(jingdian(i)|2#le#i#and#i#le#(n2):@for(jingdian(k):@for(jingdian(j):bb2(k,i,j)=@if(v2(i)#eq#j#and#v2(i-1)#eq#k,t(k)+tt(k,j),0)));b2(i)=@sum(jingdian(j):@sum(jingdian(k):bb2(k,i,j))));
@for(jingdian(k)|k#ne#1#and#k#le#n2:tv2(k)=@sum(jingdian(i)|i#le#(k):b2(i))/12+4);
@for(jingdian(k)|k#eq#1#or#k#gt#n2:tv2(k)=0);
@for(jingdian(k)|k#le#n2:@for(jingdian(j)|j#le#n:rrr(j,k)=@if(v(j)#eq#v2(k)#and#(tv2(k)-tv(j))#lt#1#and#(tv(j)-tv2(k))#lt#1,1,0)));
@for(links(j,k)|(j#gt#n#and#k#le#n2)#or#(j#le#n#and#k#gt#n)#or#(j#gt#n#and#k#gt#n):rrr(j,k)=0);
@for(jingdian(j)|j#ge#2#and#j#le#n2:rrv(j)=@sum(jingdian(k):rrr(j,k)));
@for(jingdian(k)|k#eq#1#or#k#gt#n2:rrv(k)=0);
End
```

结果（以n=5为例）：

Local optimal solution found at iteration: 5831

Objective value: 750

Variable	Value	Reduced Cost
Z	0.9500000	0.000000
N2	5.000000	0.000000
N	5.000000	0.000000
FEIYONG	960.6000	0.000000
FEIYONG2	980.4500	0.000000
V(1)	1.000000	0.000000
V(2)	0.000000	0.000000
V(3)	0.000000	0.000000
V(4)	0.000000	0.000000
V(5)	0.000000	0.000000
V(6)	0.000000	0.000000
V(7)	0.000000	0.000000
V(8)	0.000000	0.000000
V(9)	0.000000	0.000000
V(10)	0.000000	0.000000
V(11)	0.000000	0.000000

V2(1)	1.000000	0.000000
V2(2)	0.000000	0.000000
V2(3)	0.000000	0.000000
V2(4)	0.000000	0.000000
V2(5)	0.000000	0.000000
V2(6)	0.000000	0.000000
V2(7)	0.000000	0.000000
V2(8)	0.000000	0.000000
V2(9)	0.000000	0.000000
V2(10)	0.000000	0.000000
V2(11)	0.000000	0.000000

程序：（以n=5为例）

```
sets:
jingdian/1..11/:c,t,w,l,v,b,y,tv;
gailv/1..10/;
links(jingdian,jingdian):r,cc,tt;!其中：r为0-1变量（0表示两景点不相连，1表示两景点相通）；cc为两景点之间的交通费用；tt为两景点之间的交通时间;
links2(jingdian,gailv):pp,pv;
links1(jingdian,jingdian,jingdian):x,bb,ppv;
endsets
data:
t=7 24 18 12 36 30 12 9 15 24 17;!其中:t(1)=0,表示在泉城广场的逗留时间为0,因为相对旅行团而言，由于泉城广场既是起点又是终点，并未在泉城广场游玩;
c=120 423 300 135 378 390 175 90 148 303 241;!其中：c(1)=0，表示在成都的吃住费用算做，理由同上;
w= 0 0.185 0.185 0.217 0.217 0.196 0.196 0.206 0.206 0.196 0.196;
tt=0 8.54 4.74 2.82 3.44 5.08 8.4 1.32 1.54 6.14 6.6
8.54 0 1.22 11.52 12.14 10.9 13.1 8.84 8.98 14.84 15.54
4.74 1.22 0 11.22 11.82 9.38 11.58 7.66 7.46 13.44 13.9
2.82 11.52 11.22 0 0.88 7.78 8.08 4.02 4.24 5.84 6.3
3.44 12.14 11.82 0.88 0 8.42 8.24 4.66 4.88 6 6.46
5.08 10.9 9.38 7.78 8.42 0 2.18 4.24 4.04 5.98 6.74
8.4 13.1 11.58 8.08 8.24 2.18 0 6.08 6.22 3.86 2.86
1.32 8.84 7.66 4.02 4.66 4.24 6.08 0 0.3 6.28 6.74
1.54 8.98 7.46 4.24 4.88 4.04 6.22 0.3 0 6.08 6.54
6.14 14.84 13.44 5.84 6 5.98 3.86 6.28 6.08 0 2.08
6.6 15.54 13.9 6.3 6.46 6.74 2.86 6.74 6.54 2.08 0;
!其中：主对角线为零,表示各景点到自身交通费用为零;
cc=0 128 71 42 52 77 126 20 23 92 99
128 0 18 173 182 164 197 133 135 223 233
71 18 0 168 177 141 174 115 112 202 209
42 173 168 0 13 117 121 60. 64 88 95
```

```
52   182 177 13   0    126 124 70   73   90   97
76   164   141    117 126 0    33   64   61   90   101
126 197       174 121 124 33   0    91   93   58   43
20   133 115 60   70   64   91   0    5    94   101
23   135 112 64   73   61   93   5    0    91   98
92   223 202 88   90   90   58   94   91   0    31
99   233 209 95   97   101 43   101 98   31   0;
!其中：主对角线为零,表示各景点到自身的交通时间为零;
n=5;!其中：n表示计划游玩的景点数目;
pp=0.15 0.1  0.3  0.8  0.7  0.5  0.6  0.3  0.1  0
1    0.8  0.8  0.5  0.5  0.4  0.5  0.6  0.3  0.2
0.5  1    0.9  1    0.3  0    0.2  0    0.4  0.4
0.1  0    0.1  0.5  0.5  0.7  0.3  0.3  0.1  0.1
0.3  0.4  0.4  0.6  0.3  0.3  0.2  0.4  0.6  0.6
0.6  0.6  0.6  0.5  0.8  0.3  0.1  0.1  0.1  1
0.3  0.2  0.2  0.1  0    0.1  0.1  0.2  0.2  0.3
0.4  0.3  0.3  0.2  0.4  0.9  0.9  0.9  0.8  0.8
0.5  0.3  0.3  0.4  0.3  0.8  1    0.9  0.9  0.9
0.2  0.6  0.6  0.4  0.1  0    0.8  0.7  0.6  0.4
0.1  0.1  0.3  0.3  0.5  0.6  0.8  0.7  0.63      0.4;
enddata
min=0*(p-0.9)/3.3+0.9*(quan- 137)/173;!目标函数：表示计划游玩的景点数目为n时的最小费用;
quan=@sum(jingdian(j):@sum(jingdian(i):w(j)*r(i,j)*(cc(i,j)+0.45*(c(i)+c(j)))));
q=@sum(jingdian(j):@sum(jingdian(i):r(i,j)*(cc(i,j)+0.45*(c(i)+c(j)))));
@for(jingdian(i):r(i,i)=0);!约束条件：表示各景点到自身没有路线相连的约束条件;
@for(jingdian(i)|i#ge#2:@for(jingdian(j)|j#ge#2:r(i,j)+r(j,i)<1));!约束条件：表示除起点外，若旅行团从景点i到景点j去游玩（即r(i,j)=1），则不会再从景点j到景点i去游玩（即r(j,i)=0），也就是说除起点外每个景点只游玩一次;

@sum(jingdian(j):@sum(jingdian(i):r(i,j)*(tt(i,j)+0.5*(t(i)+t(j)))))<120;!约束条件:表示总的旅行时间（交通时间和景点逗留时间）不超过给定时间10天;

@for(jingdian(i):@sum(jingdian(j):r(i,j))=@sum(jingdian(j):r(j,i)));
@for(jingdian(i)|i#eq#1:@sum(jingdian(j):r(i,j))=1);
@for(jingdian(i)|i#ne#1:@sum(jingdian(j):r(i,j))<1);!这三个约束条件：表示起点有且仅有一条路线出去和一条路线进来,其它景点要么有且仅有一条路线出去和一条路线进来，要么既没有路线出去也没有路线进来;
@for(links:@bin(r));!约束条件:表示0-1变量约束;
@sum(jingdian(j):@sum(jingdian(i):r(i,j)))=n;!约束条件:表示旅游景点的数目为n的约束;
@for(jingdian(i):@for(jingdian(j)|j#gt#1#and#j#ne#i:l(j)>=l(i)+r(i,j)-(n-2)*(1-r(i,j))+(n-3)*r(j,i)));
```

```
@for(jingdian(i)|i#gt#1:l(i)<n-1-(n-2)*r(1,i);l(i)>1+(n-2)*r(i,1));!这两个约束条件：
为了控制不出现两个以上环形回路，保证有且仅有一条环形路线；
@for(jingdian(i)|1#eq#i:v(i)=1;b(i)=0);
@for(jingdian(k)|1#lt#k#and#k#le#n:@for(jingdian(i):@for(jingdian(j):x(k,i,j)=@if(0.
5#le#r(i,j)#and#r(i,j)#le#1.5#and#(i-0.5)#le#v(k-1)#and#v(k-1)#le#(i+0.5),j,0)));v(k)
=@sum(jingdian(j):@sum(jingdian(i):x(k,i,j))));
@for(jingdian(i)|2#le#i#and#i#le#(n):@for(jingdian(k):@for(jingdian(j):bb(k,i,j)=@if
(v(i)#eq#j#and#v(i-1)#eq#k,t(k)+tt(k,j),0))));
@for(jingdian(i)|2#le#i#and#i#le#(n):b(i)=@sum(jingdian(j):@sum(jingdian(k):bb(k,
i,j))));
@for(jingdian(k)|k#le#n:tv(k)=@sum(jingdian(i)|1#le#i#and#i#le#(k):b(i)/12));
!@for(jingdian(j)|j#le#n:@for(jingdian(i):ssv(i,j)=@if(v(j)#eq#i,t(i),0)));
!@for(jingdian(j)|j#le#n:sv(j)=@sum(jingdian(i):ssv(i,j)));
@for(jingdian(j)|2#le#j#and#j#le#n:@for(gailv(k):@for(jingdian(i):ppv(i,j,k)=@if(v(j)
#eq#i#and#(tv(j))#le#k#and#k#le#(tv(j)+t(i)/12),pp(i,k),0))));
@for(links2(j,k)|2#le#j#and#j#le#n:pv(j,k)=@sum(jingdian(i):ppv(i,j,k)));
@for(links2(j,k)|j#lt#2#or#j#gt#n:pv(j,k)=0);
p=@sum(links2(j,k):pv(j,k));
end
```

结果：

Variable	Value	Reduced Cost
N	5.000000	0.000000
P	1.600000	0.000000
QUAN	157.8990	0.000000
Q	927.2000	0.000000
V(1)	1.000000	0.000000
V(2)	7.000000	0.000000
V(3)	9.000000	0.000000
V(4)	8.000000	0.000000
V(5)	4.000000	0.000000
V(6)	0.000000	0.000000
V(7)	0.000000	0.000000
V(8)	0.000000	0.000000
V(9)	0.000000	0.000000
V(10)	0.000000	0.000000
V(11)	0.000000	0.000000