

栗仁武！欢迎您登录计算机软件能力认证考试系统！

2017年12月3日 星期日

准考证号：201708803262

考试剩余时间：19分17秒

姓名：栗仁武

当前时间：17:06:28

[返回试题列表](#)

问题描述

试题
编
号：

5

试题
名
称：

商路

时间
限
制：

50.0s

内存
限
制：

768.0MB

问题
描
述：

题目描述

国王小 W 总共有 n 座城市，用 1 到 n 的整数编号。除首都 1 号城市外，每座城市都有唯一的直接上级城市，并有一条道路连接二者，即这些城市和道路构成了一颗有根树。对于城市 i ，我们记它的直接上级城市为 u_i ，它到该城市道路的长度为 s_i ；同时，称 i 是 u_i 的直接下级城市。城市 i 的直接下级城市和直接下级城市的下级城市统称为城市 i 的下级城市。

每座城市的长官都想要建立一条由自己管辖的商路。对于城市 i 的长官，他管辖的商路必须形如 $a_1 a_2 \dots a_k$ ，其中 $a_1 = i$ ；对于 $1 \leq j < k$ ， a_{j+1} 是 a_j 的下级城市； k 可以是你指定的正整数。即，一条商路是从城市 i 出发，依次向下级延伸的一条路径。注意路径只用向下级城市延伸即可，不限定为直接下级。

城市 i 的价值由 v_i 和 f_i 两个参数描述。商路 $a_1 a_2 \dots a_k$ 的总价值为：

$$\sum_{j=1}^{k-1} \left\{ v_{a_j} - \left[f_{a_j} - d(a_j, a_{j+1}) \right]^2 \right\}$$

其中 $d(a_j, a_{j+1})$ 是连接 a_j 和 a_{j+1} 的道路总长度。形象地说，商路的总价值等于每小段的价值之和； v_i 和 f_i 分别描述了 i 作为起点的最理想小段价值和最佳的小段长度；小段的价值等于以它们为参数，两座城市距离为自变量的二次函数。

如有必要，一条商路允许只包含长官所在的那一座城市，即 $k = 1$ ，此时商路的总价值为 0。此外，不同城市长官建立的道路之间互相不影响，价值独立计算。

国王小 W 希望所有的 n 条商路的总价值之和最大，他现在请你帮忙。你可以替每座城市的长官决定商路的长度 k 和路径上除起点之外的城市，问最大的总价值之和是多少。

输入格式

从标准输入读入数据。

输入的第一行包含一个正整数 T ，表示数据的组数。对于测试数据，保证 $T = 10$ 。

接下来有 T 个部分，每个部分描述一组数据，其中：

每个部分第一行包含一个正整数 n ，表示城市的数量。

接下来 n 行，每行包含 4 个非负整数 u_i, s_i, v_i, f_i ，分别表示直接上级城市的编号，连接直接上级城市道路的长度，最理想小段价值和最佳的小段长度。保证： $0 \leq v_i \leq 10^{13}$ ， $0 \leq f_i \leq 10^9$ ；对于 $2 \leq i \leq n$ ，保证 $1 \leq u_i < i$ ， $1 \leq s_i \leq 10^4$ ； $u_1 = s_1 = 0$ ，仅占位无实际意义（首都没有上级城市）。

输出格式

输出到标准输出。

输出 T 行，每行一个整数，表示相应数据中 n 条商路最大的总价值。因为这个数可能很大，你只需要输出这个数除以 10^{18} 的余数。

子任务

各组数据特点如下所示：

测试点	$n \leq$	其他特点
1,2,3	10	无
4,5,6	10^2	
7,8,9	2,000	
10,11,12	10^5	对于 $2 \leq i \leq n, u_i = i - 1$
13,14,15		对于 $2 \leq i \leq n, u_i = \lfloor \frac{u_i}{2} \rfloor$
16,17,18,19,20		无

其中， $\lfloor \frac{a}{b} \rfloor$ 表示对 $\frac{a}{b}$ 下取整，即等于 a 整除 b 。

样例 1 输入

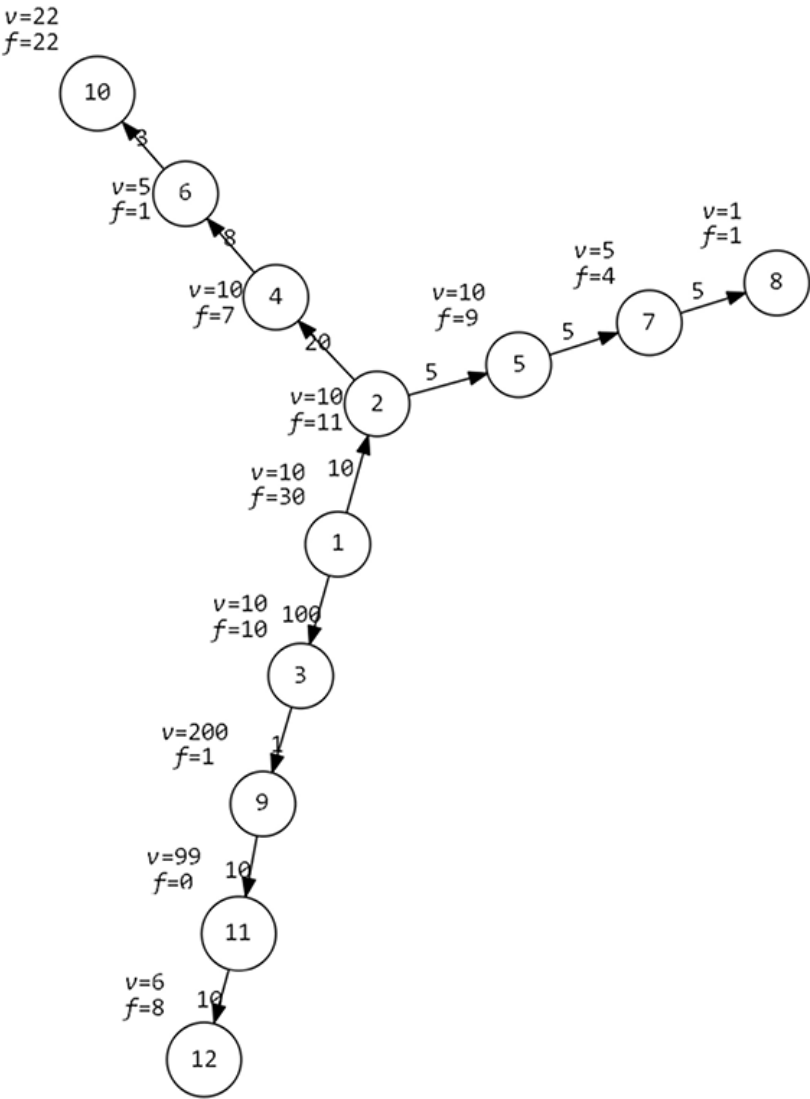
```
1
12
0 0 10 30
1 10 10 11
1 100 10 10
2 20 10 7
2 5 10 9
4 8 5 1
5 5 5 4
7 5 1 1
3 1 200 1
6 3 22 22
9 10 99 0
11 10 6 8
```

样例 1 输出

224

样例 1 解释

这个例子的城市及道路如下图所示（图中箭头表示的是上下级关系）：



各城市的长官管辖的商路如下：

城市 1：1 → 4 → 6 → 10，总收益 20。

城市 2：2 → 7 → 8，总收益 13。

城市 3: $3 \rightarrow 9 \rightarrow 11$, 总收益 48。

城市 4: $4 \rightarrow 6 \rightarrow 10$, 总收益 10。

城市 5: $5 \rightarrow 8$, 总收益 9。

城市 6: $6 \rightarrow 10$, 总收益 1。

城市 7: $7 \rightarrow 8$, 总收益 4。

城市 8: 8, 总收益 0。

城市 9: $9 \rightarrow 11$, 总收益 119。

城市 10: 10, 总收益 0。

城市 11: 11, 总收益 0。

城市 12: 12, 总收益 0。

对于城市 1, $a_1 = 1$, $a_2 = 4$, $a_3 = 6$, $a_4 = 10$ 。收益为:

$$\begin{aligned} & \{v_1 - [f_1 - d(1,4)]^2\} + \{v_4 - [f_4 - d(4,6)]^2\} + \{v_6 - [f_6 - d(6,10)]^2\} \\ &= [10 - (30 - 30)^2] + [10 - (7 - 8)^2] + [5 - (1 - 3)^2] = 20 \end{aligned}$$

样例 2 输入

```
10
10
0 0 981021 2878
1 2982 103544 16423
2 9309 1146606 60
1 9632 339699 4022
1 4859 1050430 27644
4 7152 1016794 8381
2 691 202924 1579
5 682 312623 1947
1 5622 434383 1966
1 1036 337962 4867
10
0 0 613160 4178
1 6039 530965 36077
1 2641 588435 10697
2 8015 1773940 21360
2 9456 1536141 20760
3 1540 1069223 6060
3 4132 1802960 9712
4 1295 338084 630
```

4 5977 858393 3942
5 1981 1569807 8875
10
0 0 3297948 29281
1 2838 4499071 56444
2 1723 5082712 58167
3 6932 1899951 86996
4 6111 4673596 39622
5 4854 1283153 52866
6 1600 3076850 58518
7 2887 3936856 44586
8 9662 1653482 1115
9 6359 1438146 7533
10
0 0 3827584 9067
1 1114 1938309 90351
2 1913 3135879 31961
3 210 3118784 55465
4 6982 1115456 4482
5 4326 2504229 8147
6 3567 1296569 21072
6 857 3723988 867
7 2726 1349738 682
6 2569 1465079 9994
10
0 0 4152770 20839
1 3728 1993622 104795
2 3358 1563274 31216
3 6794 2231093 43445
4 2197 2316621 7602
5 1174 2985157 7136
6 8333 103275 6346
6 5129 3083322 3511
7 469 2368248 9692
7 5732 3765843 8563
10
0 0 3091416 127451
1 2323 3358598 10828
2 8139 3723827 15116
3 9725 2486973 25964
4 1954 2821451 18368
5 8257 3938729 32443
6 2554 4006939 13576
6 5881 511102 9605
7 7999 1954660 1148
7 3260 1916815 8710
10
0 0 366406 31967
1 7426 2566312 2237
1 654 2294113 5913

```

2 8186 893528 10662
2 6453 1512505 13321
4 6429 616635 6615
6 5842 2578315 14449
5 3568 2988564 9781
6 1704 1044306 9036
7 4747 155934 4217
10
0 0 2481887 67676
1 2514 2068792 41590
2 5134 2460668 181
1 608 1693903 5729
2 2202 2804903 37359
5 611 1673964 6522
6 429 1245126 10648
6 7898 358662 6483
7 4194 1315545 8960
6 1193 224838 7332
10
0 0 3403835 116677
1 77 2005502 44152
2 2913 3325129 47250
2 8094 2319341 37
3 3750 458636 52717
5 8465 1635392 21518
6 9440 1757959 2079
6 8555 2306608 4397
6 5530 2658501 440
7 1594 1833614 690
10
0 0 1477756 44072
1 8380 3425538 18191
2 7951 978970 6357
1 2602 184752 6671
3 725 2947560 1525
5 1314 3589295 39573
6 1755 1675874 6532
6 8848 2470638 3789
7 1620 2129738 9070
7 9509 2363613 6569

```

样例 2 输出

```

970205
613151
0
4682720
3959657
2626087

```

19106
0
1522734
5999689

样例输入

```
1
12
0 0 10 30
1 10 10 11
1 100 10 10
2 20 10 7
2 5 10 9
4 8 5 1
5 5 5 4
7 5 1 1
3 1 200 1
6 3 22 22
```



```
9 10 99 0
11 10 6 8
```

样例输出

```
224
```

样例输入

```
10
10
0 0 981021 2878
1 2982 103544 16423
2 9309 1146606 60
1 9632 339699 4022
1 4859 1050430 27644
4 7152 1016794 8381
2 691 202924 1579
5 682 312623 1947
1 5622 434383 1966
1 1036 337962 4867
10
0 0 613160 4178
1 6039 530965 36077
1 2641 588435 10697
2 8015 1773940 21360
2 9456 1536141 20760
3 1540 1069223 6060
3 4132 1802960 9712
4 1295 338084 630
4 5977 858393 3942
5 1981 1569807 8875
10
0 0 3297948 29281
1 2838 4499071 56444
2 1723 5082712 58167
3 6932 1899951 86996
4 6111 4673596 39622
5 4854 1283153 52866
6 1600 3076850 58518
7 2887 3936856 44586
8 9662 1653482 1115
9 6359 1438146 7533
10
0 0 3827584 9067
1 1114 1938309 90351
2 1913 3135879 31961
3 210 3118784 55465
4 6982 1115456 4482
5 4326 2504229 8147
6 3567 1296569 21072
6 857 3723988 867
7 2726 1349738 682
6 2569 1465079 9994
10
0 0 4152770 20839
1 3728 1993622 104795
2 3358 1563274 31216
3 6794 2231093 43445
4 2197 2316621 7602
5 1174 2985157 7136
6 8333 103275 6346
6 5129 3083322 3511
7 469 2368248 9692
7 5732 3765843 8563
10
0 0 3091416 127451
1 2323 3358598 10828
2 8139 3723827 15116
3 9725 2486973 25964
4 1954 2821451 18368
5 8257 3938729 32443
6 2554 4006939 13576
6 5881 511102 9605
7 7999 1954660 1148
7 3260 1916815 8710
10
0 0 366406 31967
1 7426 2566312 2237
1 654 2294113 5913
2 8186 893528 10662
```

```
2 6453 1512505 13321
4 6429 616635 6615
6 5842 2578315 14449
5 3568 2988564 9781
6 1704 1044306 9036
7 4747 155934 4217
10
0 0 2481887 67676
1 2514 2068792 41590
2 5134 2460668 181
1 608 1693903 5729
2 2202 2804903 37359
5 611 1673964 6522
6 429 1245126 10648
6 7898 358662 6483
7 4194 1315545 8960
6 1193 224838 7332
10
0 0 3403835 116677
1 77 2005502 44152
2 2913 3325129 47250
2 8094 2319341 37
3 3750 458636 52717
5 8465 1635392 21518
6 9440 1757959 2079
6 8555 2306608 4397
6 5530 2658501 440
7 1594 1833614 690
10
0 0 1477756 44072
1 8380 3425538 18191
2 7951 978970 6357
1 2602 184752 6671
3 725 2947560 1525
5 1314 3589295 39573
6 1755 1675874 6532
6 8848 2470638 3789
7 1620 2129738 9070
7 9509 2363613 6569
```

样例输出

```
970205
613151
0
4682720
3959657
2626087
19106
0
1522734
5999689
```

答题栏

试题编号: 5

试题名称: 商路

编译环境: 1. Java ▼

答案程序:

提交确认:

- 以下必须全部满足才能提交:
- ☐ 我的程序没有使用package语句来定义包的信息。（如果定义了将无法评测）
 - ☐ 我的程序的主类名称是Main，主类的定义是public class Main。我的程序运行的入口是主类Main中的main函数，定义为public static void main(String[] args)。
 - ☐ 我的程序是从标准输入(System.in)中读入数据的（使用Scanner等类来处理System.in的输入视为满足条件），结果是输出到标准输出(System.out)的。
 - ☐ 我的程序是从标准输入中读入数据的，结果是输出到标准输出的。
 - ☐ 我的程序中没有“请输入n”之类的输入输出提示，也没有输出中间的结果，所有的输出内容都与题设中的输出格式相对应。我的输出格式（包括换行和大小写等）与题设中输出格式的要求相符。

提 交

查看上一次提交

返回试题列表