

NOI2015 浙江省组队选拔赛试题

第二试

题目名称	黑客技术	醉醺醺的幻想乡	幻想乡 wifi 搭建计划
可执行文件名	lock	mincost	wifi
输入文件名	lock.in	mincost.in	wifi.in
输出文件名	lock.out	mincost.out	wifi.out
每个测试点时限		3	3
内存限制		256M	256M
测试点数目	10	10	10
每个测试点分值	10	10	10
是否有部分分	有	否	否
题目类型	提交答案型	传统型	传统型
是否有附加文件	有	无	无

提交源程序须加后缀

对于 C++ 语言		mincost.cpp	wifi.cpp
对于 C 语言		mincost.c	wifi.c
对于 Pascal 语言		mincost.pas	wifi.pas

注意：最终测试时，所有编译命令均不打开任何优化开关。

Problem A. 黑客技术

Input file: lock.in
Output file: lock.out
Time limit: 3s
Memory limit: 256MB

最近傲娇少女幽香很忙，于是让跳蚤国王出了一道题给 ZJOI2015，一时间人人自危。你秘密潜伏了下来想窃取跳蚤国王的试题。

终于有一天，趁跳蚤国王出去练习跳高，你偷偷打开了跳蚤国王的电脑——结果发现试题被 10 个电子密码锁锁住了！

任何事情都阻挡不了你进省队的决心！每个电子密码锁内部有 10 道检测工序，输入密码后，每通过一道检测工序密码锁就会输出一行“ok”。如果输出了 10 行“ok”那么就解开了该密码锁。

根据常识知，跳蚤国王一定是用高级计算机语言 fleaplus 写代码的，然后用 fleaplus 的编译器编译成汇编代码，再用汇编编译器编译为机器码。

显然你没有这 10 个电子密码锁的 fleaplus 源代码，通过反汇编，你获得了这 10 个电子密码锁的汇编代码。

跳蚤的程序开始运行时会开出长度为 2^{23} 的 32 位有符号整数数组 a ，元素分别为 $a_0, a_1, \dots, a_{2^{23}-1}$ ，并全部初始化为 0。

在跳蚤的汇编语言中，有三种量：“@ c ”，“\$ c ”，“# c ”（不含引号）。其中 c 是个整数且 $c \in [-2^{31}, 2^{31})$ 。

- “@ c ” 表示整数常量 c 。
- “\$ c ” 表示 a_c 。
- “# c ” 表示 a_{a_c} 。

如果程序运行过程中试图访问的量会导致数组下标越界，那么就会产生 Runtime Error 错误并异常退出。

汇编代码由若干行组成，每行是一条语句（从 1 开始编号），程序会从第 1 条语句从上往下执行。

语句有下面这么几种：

- 赋值语句：= a b
 - a, b 是某个量。
 - 表示把 b 的值赋给 a 。
 - 如果 a 是个整数常量那么会产生 Runtime Error 错误并异常退出。
- 输入语句：getc c 和 geti c
 - c 是某个量。
 - getc c 表示读入一个字符，把该字符的 ASCII 码存入 c 中。

- `geti c` 表示读到下一个非空白字符的位置，读入十进制表示的 32 位有符号整数，存入 `c` 中。（空白字符即 ASCII 码为 9,10,13,32 的字符）
- 如果 `c` 是个整数常量那么会产生 `Runtime Error` 错误并异常退出。如果读入整数失败则输出相关信息并异常退出。
- 输出语句: `putc c` 和 `puti c`
 - `c` 是某个量。
 - `putc c` 表示输出一个字符，`c` 是该字符的 ASCII 码。
 - `puti c` 表示输出整数 `c` 的十进制表示。
- 条件跳转语句: `if c t`
 - `c,t` 是某个量。
 - 表示如果 `c` 不是 0，那么跳转到第 `t` 条语句继续执行（即下一条要执行的语句变为第 `t` 条语句）；
 - 如果是 0，那么不进行任何操作，继续执行下一条语句。
- 运算语句: `o a b c`
 - `a,b,c` 是某个量，`o` 是某个运算符。
 - 该语句表示把 `a` 和 `b` 进行 `o` 的运算，把结果赋值给 `c`。
 - 如果 `o` 是 `+` 则表示加法，即 $a + b$ 。
 - 如果 `o` 是 `-` 则表示减法，即 $a - b$ 。
 - 如果 `o` 是 `*` 则表示乘法，即 $a \times b$ 。
 - 如果 `o` 是 `/` 则表示做除法然后下取整，即 $\lfloor \frac{a}{b} \rfloor$ 。
 - 如果 `o` 是 `<` 则表示 $a < b$ 则结果为 1，否则为 0。
 - 如果 `o` 是 `==` 则表示 $a = b$ 则结果为 1，否则为 0。
 - 如果 `c` 是个整数常量或者作除法时除数为 0 那么会产生 `Runtime Error` 错误并异常退出。

程序运行过程中如果试图执行第 `-1` 条语句那么程序正常结束，如果试图执行除此之外的不存在的语句那么会产生 `Runtime Error` 错误并异常退出。

如果执行的语句条数超过 10^7 那么会产生 `Time Limit Exceeded` 错误并异常退出。

现在，是时候展现你卓越的黑客技术了！

Input

本题为提交答案题。所有输入数据 `lock1.in~lock10.in` 已放置在选手目录下。

每个输入文件表示一个密码锁的汇编代码。

我们还提供了一个来自跳蚤国的模拟器，能直接执行汇编代码。

在终端中先切换到该试题的目录下（Windows 选手请使用 `cmd`）。

然后在终端中运行：`./run <code>`。其中 `<code>` 是你执行的汇编代码的文件名。例如：`./run lock1.in`（Windows 选手请使用 `run_win32 lock1.in`）

模拟器将从终端中读入数据。如果你想从文件中读入，请使用 `<` 来指定。

例如：`./run lock1.in <lock1.out`

Output

针对给定的 10 个输入文件 `lock1.in~lock10.in`，你需要分别给出你的输出文件 `lock1.out~lock10.out`。

每个输出文件包含一些字符，表示你输入的密码。

Example

见 `lock0.in` 与 `lock0.out`。

该程序会读入一个整数，如果这个整数是 666 那么就会输出 10 行 “ok”。

Scoring

每个密码锁为一个测试点，每个测试点 10 分。

对于每个密码锁，我们会以你的输出文件作为输入运行密码锁。

如果程序异常退出则该测试点 0 分。

否则，我们会检查密码锁输出的每一行，设有 x 行为 “ok”。如果 $x \geq 10$ 则该测试点满分，否则该测试点得 x 分。

Postscript

你花了很长时间破解密码锁，最后也没有破解出来。最后硬着头皮去了 ZJOI2015，发现自己的事迹被写在了题面里。

Note

题目中提到的“32 位有符号整数”等价于 C/C++ 中的 `int`，Pascal 中的 `longint`，能表示 $[-2^{31}, 2^{31})$ 以内的整数。计算过程可能会溢出，比如： $2147483647 + 1 = -2147483648$ ， $100000 \times 100000 = 1410065408$ 。

另附 ASCII 字符对照表：

ASCII	字符	ASCII	字符	ASCII	字符	ASCII	字符
0	NUT	32	(space)	64	@	96	`
1	SOH	33	!	65	A	97	a
2	STX	34	"	66	B	98	b
3	ETX	35	#	67	C	99	c
4	EOT	36	\$	68	D	100	d
5	ENQ	37	%	69	E	101	e
6	ACK	38	&	70	F	102	f
7	BEL	39	'	71	G	103	g
8	BS	40	(72	H	104	h
9	HT	41)	73	I	105	i
10	LF	42	*	74	J	106	j
11	VT	43	+	75	K	107	k
12	FF	44	,	76	L	108	l
13	CR	45	-	77	M	109	m
14	SO	46	.	78	N	110	n
15	SI	47	/	79	O	111	o
16	DLE	48	0	80	P	112	p
17	DC1	49	1	81	Q	113	q
18	DC2	50	2	82	R	114	r
19	DC3	51	3	83	X	115	s
20	DC4	52	4	84	T	116	t
21	NAK	53	5	85	U	117	u
22	SYN	54	6	86	V	118	v
23	TB	55	7	87	W	119	w
24	CAN	56	8	88	X	120	x
25	EM	57	9	89	Y	121	y
26	SUB	58	:	90	Z	122	z
27	ESC	59	;	91	[123	{
28	FS	60	<	92	\	124	
29	GS	61	=	93]	125	}
30	RS	62	>	94	^	126	~
31	US	63	?	95	_	127	DEL

Problem B. 醉醺醺的幻想乡

Input file: mincost.in
Output file: mincost.out
Time limit: 3s
Memory limit: 256MB

傲娇少女幽香是一个很萌很萌的妹子，这些天幻想乡的大家都不知道为何开始拼命喝酒。很快酒就供不应求了，为了满足大家的需求，幽香决定在森林里酿酒。

经过调查，幽香发现在森林里面有一些地方非常适合酿酒，有一些地方则非常适合存酒。

幽香把这些适合酿酒的地方称为酿酒点，不妨认为有 n 个酿酒点，从1到 n 标号。

同时也有 m 个适合存酒的地方，幽香将它们称为存酒点，从1到 m 标号。

在一些酿酒点和存酒点之间存在通道，如果酿酒点 i 到存酒点 j 之间存在通道，那么 i 生产的酒就可以被运输到 j 。

但是在一个地方酿酒是需要消耗幽香的魔力的，由于存在管理上的因素，在酿酒点 i ，制造 x 升的酒，需要花费 $a_i x^2 + b_i x$ 的魔力，注意 x 不一定是一个非负整数，也可以是一个非负实数，同时在这个点最多只能制造 c_i 升的酒。

每个存酒点 j 有一个容量 d_j ，表示这个存酒点最多能存多少升的酒。幽香打算存尽量多的酒。由于在森林中只有酿酒点可以酿酒，也只有存酒点可以存酒，那么她需要在一些酿酒点生产一些酒并且通过通道将酒运送到存酒点。

当然幽香想要节省自己的魔力，所以想让你帮忙算出在满足要求的情况下，最少花费的魔力是多少？

Input

第一行两个正整数 n, m ，表示酿酒点和存酒点的数量。

接下来 n 行，第 i 行三个数 a_i, b_i, c_i ，表示在酿酒点 i 制造酒的花费的系数，和上限。

接下来一行 m 个数，依次为每个酿酒点的 d_i 值。

接下来 n 行，每行 m 个数，其中第 i 行第 j 个表示酿酒点 i 到存酒点 j 有没有通道(1表示有，0表示没有)。

Output

输出第一行表示幽香最多能存多少升酒（注意这肯定是个整数，直接输出即可）。

输出一行表示最小花费魔力，注意这肯定是一个有理数，化简后按照 a/b 的形式输出(0输出0/1)。

Constraints

对于30%的数据：所有 $a_i = 0$ 。

对于另30%的数据：最终答案的分母 ≤ 1000 。

对于100%的数据： $1 \leq n \leq 100, 1 \leq m \leq 100$ 。

对于所有数据， $0 \leq a_i, b_i, c_i, d_i \leq 3$ 且都是整数。同时对于每个 i ， $a_i + b_i > 0$ ，通道的数量不超

过500条。

非常神奇的是，对于所有数据存在一个正整数 $X \leq 10^{12}$ ，使得存在一个最优解，使得所有路径上运送的酒的体积都是 $1/X$ 升的倍数。

Example

mincost.in	mincost.out
10 10	8
0 2 3	42/1
2 3 2	
3 1 3	
1 2 1	
1 0 1	
1 1 0	
3 3 0	
1 2 2	
3 1 1	
3 1 0	
3 1 2 2 3 1 1 2 2 0	
0 0 0 0 0 0 0 1 0 0	
0 0 0 0 0 0 0 0 0 0	
0 0 0 0 1 0 0 0 0 0	
0 0 0 1 0 0 0 0 0 0	
1 0 0 0 1 0 0 0 0 0	
1 0 1 0 0 0 0 1 1 0	
0 0 0 0 0 0 0 0 0 0	
0 0 0 0 0 0 0 0 0 0	
0 0 1 0 0 0 1 0 0 0	
0 0 0 0 0 1 0 0 1 0	

Problem C. 幻想乡wifi搭建计划

Input file: wifi.in
Output file: wifi.out
Time limit: 3s
Memory limit: 256MB

傲娇少女幽香是一个很萌很萌的妹子。随着科技的进步，幻想乡的大家也开始使用手机了。这时幽香发现没人来她的太阳花田玩了。她感到很伤心，于是向别人打听了一下，才知道原来大家都嫌弃这里没有wifi，手机上网还需要流量。

怎么办呢，幽香决定赶快搭建几个wifi点，让所有人都能在太阳花田里畅快地上网。

我们可以近似地把太阳花田看成一个 y 轴在 $[0, R]$ 之间， x 坐标为 $(-\infty, +\infty)$ (也就是在 x 轴上无限延伸)的无限长方形。

太阳花田里里面有 n 个景点，是游客们经常光顾的，幽香认为只要让这些景点尽量被wifi覆盖，那么游客们就肯定心满意足了。

八云紫表示她可以帮助幽香架设wifi路由器。现在通用的路由器，每个的覆盖半径正好也是 R 。八云紫扫视了一遍地图，发现在太阳花田的外面，只有 m 个有网络的地点，她只可以在那里架设路由器。如果你在点 p 搭建了路由器，那么位于 q 的地点，只要 p 和 q 的欧几里得距离小于等于 R ， q 点就会被wifi覆盖。

同时八云紫表示，架设难度随着地点的不同而不同，所以收费也不一样，在第 i 个位置架设需要 c_i 的钱。

现在幽香想要覆盖尽量多的景点，在这个前提下，幽香也想要尽量节省钱。你能帮助她吗？

Input

输入第一行三个正整数 n, m, R ，满足 $1 \leq R \leq 10^8$ ，分别表示景点的数量，网络架设地点的数量和太阳花田的宽度。

接下来 n 行，每行两个整数 x, y ，满足 $-10^8 \leq x \leq 10^8, 0 \leq y \leq R$ ，表示一个景点。两个景点的位置不会重合。

接下来 m 行，每行三个整数 x, y, c ，满足 $-10^9 \leq x \leq 10^9, -10^8 < y < 0$ 或者 $R < y < 10^8, 0 \leq c \leq 10^4$ 。表示一个网络架设点的位置和花费。两个网络架设点不会重合。

Output

输出第一行表示最多覆盖的景点数。

第二行表示在覆盖景点最多的前提下，最少的花费。

Constraints

对于10%的数据， $n, m \leq 20$ 。

对于另30%的数据， $n, m \leq 100$ ，所有网络架设点的 y 坐标都 $> R$ 。

对于另60%的数据， $n, m \leq 100$ 。

Example

wifi.in	wifi.out
10 10 10000	10
6743 2963	10438
3505 1986	
3565 7235	
1735 5522	
16877 5597	
11621 6	
3100 8243	
1750 6173	
5709 7671	
7915 3915	
14339 -438 3075	
4278 15210 8371	
13996 19000 6750	
17049 -4969 7788	
737 16339 2934	
904 14023 2322	
8982 14759 4311	
13102 11458 5554	
4135 12183 576	
5087 -2459 6787	