

2017國中生暑期程式設計營

--練習賽--

本次競賽共有11題

- 有8題基礎題目與3題進階題目
- 進階題目可能需要用到一些資料結構或者演算法才能解題
- 如果解題效率太差，會獲得TLE，請思考更好的解法
- **使用cin,cout的人請在第一行加入以下程式碼避免「TLE」**
- **`ios::sync_with_stdio(0); cin.tie(0);`**
- **建議：每一份code最前面都先放，以免不必要的TLE。**
- 輸出請完全按照題目要求，多換行或少換行都有可能得到WA
- 每題皆有部分分
- 每題的滿分不一定相同，如下表
- 比賽時間為3小時
- 比賽結束前一小時凍結記分板
- 比賽期間除工作人員及隊友外，不得與其他人交談
- 比賽期間將會關閉對外網路
- 競賽系統上的參考資料可以使用
- 如對題目有任何問題，請使用系統的詢問功能提問
- 此份題目皆沒有多測資輸入

題目名稱	出題者	Time(s)	Mem(mb)	分數
題目A--咖波說你好	<i>Penguin1124</i>	5	130	100
題目B--給我好好關燈	<i>tangierne1202</i>	0.5	100	200
題目C--古董計算機	<i>ForeverGa</i>	0.5	100	200
題目D--肉肉搜尋器	<i>roger0999</i>	1	100	200
題目E--神奇的秘笈	<i>我姓潘</i>	1	100	300
題目F--箭頭	<i>tangierne1202</i>	0.5	100	300
題目G--想吃東西的咖波	<i>lys0829</i>	0.5	100	300
題目H--小雞們	<i>lys0829</i>	2	100	400
題目I--肉肉在哪裡	<i>MiohitoKiri5474</i>	3.9	256	500
題目J--XOR	<i>minson123</i>	0.8	100	500
題目K--大樹	<i>minson123</i>	1	100	500

A 咖波說你好

TimeLimit: 5s

MemLimit: 130MB

題目敘述



某日，閒閒沒事的咖波跑到TFCis台南一中資訊社了。喜歡咖波喜歡到快要敗一隻回來當社寵的T19們當然想要好好認識他，紛紛報上自己的名號，咖波一時之間都快要反應不過來了！趕緊喊停而且威脅再吵就要吃掉他們，並要他們發揮他們的專業---寫個程式來幫助咖波記住名字吧！

但是要注意---如果你錯了3次，咖波就會生氣不理你了。

輸入說明

首先有一個數字 n ，代表有 n 個人，接著輸入 n 個字串 a ，代表每個人的名字。

$0 \leq n \leq 10^7$ ， a 的長度小於25，由大小寫字母組成。

輸出說明

請輸出"Hello [name1], [name2], ... , [namen]!"(不包含雙引號)。

結尾請換行。

範例

Input	Output
3 tangerine1202 Ga Minson123	Hello tangerine1202, Ga, Minson123!

配分

額外限制	分數
$n \leq 100$ && 字串長度=5	10
$n \leq 1000$ && 字串長度=10	10
$n \leq 10000$ && 字串長度=15	10
$n \leq 100000$ && 字串長度=20	15
$n \leq 1000000$ && 字串長度=25	15
$n \leq 10000000$ && 字串長度=15	20
$n = 10000000$ && 字串長度=10	20

B 給我好好關燈

TimeLimit: 1s

MemLimit: 100MB

題目敘述

節能減碳是個好習慣，至少咖波是這麼認為的。

但是好巧不巧潘先生有個非常糟糕的習慣，只要看到有燈沒開，都會手很賤的把它開起來。

這麼愛護地球的咖波自然不會放任他這樣一直浪費電下去，所以都會特地跑去關燈。（到底是誰這麼過分在虐待咖波的啊啊啊啊啊啊啊啊！）

但是總務MiohioKiri5474非常頭痛，因為電燈開開關關的，總有一些耗材是會一直耗損的，於是他裝了一個儀器紀錄開關一共被使用了幾次。

很不巧，這些記錄檔太難分析了，正在忙著打VG的Miohitokiri5474只好請你幫忙了。

已知一開始的燈都是關著的，所以當電燈開關每被按一下，第一次代表開，第二次代表關，依此類推。

因為最終只需要留一盞燈，可憐已經沒有肝的副社長ForeverGa要去那邊工作，所以請你輸出那邊有燈，讓他可以直接去那邊工作吧。

輸入說明

第一行會有一個數字 n ($1 \leq n \leq 100$)，代表有 n 個記錄檔。

每個記錄檔會一個數字 m ，代表這個紀錄檔有 m 筆資料。

接下來會有 m 個數字 a_0 到 a_{m-1} ，為燈泡編號。（ a 屬於int範圍）

輸出說明

請輸出最後哪個燈是開著的，保證只剩一顆燈泡是開著的。

範例

Input	Output
2	3
5	1
-1 2 3 -1 2	
3	
0 1 0	

範例說明

第一行 2 代表有 2 個記錄檔，
 第二行 5 代表第一筆紀錄檔有 5 筆資料，
 接下來 5 個數字分別代表燈泡的開關過程：

- 1 號燈泡被打開、
- 2 號燈泡被打開、
- 3 號燈泡被打開、
- 2 號燈泡被關掉、
- 1 號燈泡被關掉。

最後剩下 3 是開著的，所以輸出 3。

第二筆紀錄檔以此類推，輸出 1。

配分

額外限制	分數
$0 \leq a \leq 200000$	25
$-200000 \leq a \leq 200000$	50
題目範圍	125

C 古董計算機

TimeLimit: 0.5s

MemLimit: 100MB

題目敘述

因為TFcis的社部才剛整修完，所以還有些東西還沒有好好整理完，堆在後面。

咖波最喜歡尋寶了！找到一個寶物可是他最喜歡的一件事呢。因此他跑去後面那堆垃圾裡面翻翻找找，還真的給他翻出東西來了。那是一台古董計算機，但是眼尖的咖波發現了它與其他計算機的不同——它是在計算字元用的！太神奇了，咖波忍不住與爆肝（~~喔不他已經沒有肝了~~）中的副社長大人ForeverGa討論，但是你也知道她現在都在爆肝ing了自然沒有時間去處理咖波的問題，然而你現在正好手邊沒事，就幫幫我們爆肝的副社長吧。

輸入說明

首先會有一個整數n，代表會有n筆操作。

每筆操作中，會先給一個被運算字元a。接下來會有三次運算，每次運算給一個運算符號x (+, -, *, /, %)和一個運算字元c，請將a與c做x運算。

$0 < n < 50$

'!'<= a, c <='~'

小叮嚀：運算過程可能超出char的儲存範圍

輸出說明

輸出每筆操作的運算結果，每次輸出之後請換行，應有n筆結果。

範例輸入

Input	Output
3 1 * s % Y + V T /) + W % { \ * J /\n % j	r Y J

配分

額外限制	分數
題目限制	200

The ASCII code
 www.theasciicode.com.ar

American Standard Code for Information Interchange

ASCII control characters				ASCII printable characters								Extended ASCII characters												
DEC	HEX	Simbolo ASCII		DEC	HEX	Simbolo	DEC	HEX	Simbolo	DEC	HEX	Simbolo	DEC	HEX	Simbolo	DEC	HEX	Simbolo	DEC	HEX	Simbolo	DEC	HEX	Simbolo
00	00h	NULL (carácter nulo)		32	20h	espacio	64	40h	@	96	60h	`	128	80h	Ç	160	A0h	á	192	C0h	À	224	E0h	Ó
01	01h	SOH (inicio encabezado)		33	21h	!	65	41h	A	97	61h	a	129	81h	Ù	161	A1h	â	193	C1h	Á	225	E1h	Ô
02	02h	STX (inicio texto)		34	22h	"	66	42h	B	98	62h	b	130	82h	Ê	162	A2h	ã	194	C2h	Â	226	E2h	Õ
03	03h	ETX (fin de texto)		35	23h	#	67	43h	C	99	63h	c	131	83h	Ë	163	A3h	ä	195	C3h	Ã	227	E3h	Ö
04	04h	EOT (fin transmisión)		36	24h	\$	68	44h	D	100	64h	d	132	84h	Ì	164	A4h	å	196	C4h	Ä	228	E4h	Ø
05	05h	ENQ (enquiry)		37	25h	%	69	45h	E	101	65h	e	133	85h	Í	165	A5h	æ	197	C5h	Å	229	E5h	Ù
06	06h	ACK (acknowledgement)		38	26h	&	70	46h	F	102	66h	f	134	86h	Î	166	A6h	ç	198	C6h	Æ	230	E6h	Þ
07	07h	BEL (timbre)		39	27h	'	71	47h	G	103	67h	g	135	87h	Ï	167	A7h	¸	199	C7h	Ç	231	E7h	ß
08	08h	BS (retroceso)		40	28h	(72	48h	H	104	68h	h	136	88h	Ĳ	168	A8h	ˆ	200	C8h	¸	232	E8h	Ü
09	09h	HT (tab horizontal)		41	29h)	73	49h	I	105	69h	i	137	89h	Ĳ	169	A9h	˜	201	C9h	ˆ	233	E9h	Ý
10	0Ah	LF (salto de linea)		42	2Ah	*	74	4Ah	J	106	6Ah	j	138	8Ah	Ĳ	170	AAh	¸	202	CAh	˜	234	EAh	ÿ
11	0Bh	VT (tab vertical)		43	2Bh	+	75	4Bh	K	107	6Bh	k	139	8Bh	Ĳ	171	ABh	½	203	CBh	˜	235	EBh	ÿ
12	0Ch	FF (form feed)		44	2Ch	,	76	4Ch	L	108	6Ch	l	140	8Ch	Ĳ	172	ACH	¼	204	CDh	˜	236	ECh	ÿ
13	0Dh	CR (retorno de carro)		45	2Dh	-	77	4Dh	M	109	6Dh	m	141	8Dh	Ĳ	173	ADh	⅓	205	CDh	˜	237	EDh	ÿ
14	0Eh	SO (shift Out)		46	2Eh	.	78	4Eh	N	110	6Eh	n	142	8Eh	Ĳ	174	AEd	⅔	206	CEh	˜	238	Eh	ÿ
15	0Fh	SI (shift In)		47	2Fh	/	79	4Fh	O	111	6Fh	o	143	8Fh	Ĳ	175	AFh	¸	207	CFh	˜	239	Fh	ÿ
16	10h	DLE (data link escape)		48	30h	0	80	50h	P	112	70h	p	144	90h	Ĳ	176	B0h	ˆ	208	D0h	˜	240	F0h	ÿ
17	11h	DC1 (device control 1)		49	31h	1	81	51h	Q	113	71h	q	145	91h	Ĳ	177	B1h	ˆ	209	D1h	˜	241	F1h	±
18	12h	DC2 (device control 2)		50	32h	2	82	52h	R	114	72h	r	146	92h	Ĳ	178	B2h	ˆ	210	D2h	˜	242	F2h	¼
19	13h	DC3 (device control 3)		51	33h	3	83	53h	S	115	73h	s	147	93h	Ĳ	179	B3h	ˆ	211	D3h	˜	243	F3h	½
20	14h	DC4 (device control 4)		52	34h	4	84	54h	T	116	74h	t	148	94h	Ĳ	180	B4h	ˆ	212	D4h	˜	244	F4h	¾
21	15h	NAK (negative acknowle.)		53	35h	5	85	55h	U	117	75h	u	149	95h	Ĳ	181	B5h	ˆ	213	D5h	˜	245	F5h	¸
22	16h	SYN (synchronous idle)		54	36h	6	86	56h	V	118	76h	v	150	96h	Ĳ	182	B6h	ˆ	214	D6h	˜	246	F6h	ˆ
23	17h	ETB (end of trans. block)		55	37h	7	87	57h	W	119	77h	w	151	97h	Ĳ	183	B7h	ˆ	215	D7h	˜	247	F7h	ˆ
24	18h	CAN (cancel)		56	38h	8	88	58h	X	120	78h	x	152	98h	Ĳ	184	B8h	ˆ	216	D8h	˜	248	F8h	ˆ
25	19h	EM (end of medium)		57	39h	9	89	59h	Y	121	79h	y	153	99h	Ĳ	185	B9h	ˆ	217	D9h	˜	249	F9h	ˆ
26	1Ah	SUB (substitute)		58	3Ah	:	90	5Ah	Z	122	7Ah	z	154	9Ah	Ĳ	186	BAh	ˆ	218	DAh	˜	250	FAh	ˆ
27	1Bh	ESC (escape)		59	3Bh	;	91	5Bh	[123	7Bh	{	155	9Bh	Ĳ	187	BBh	ˆ	219	DBh	˜	251	FBh	ˆ
28	1Ch	FS (file separator)		60	3Ch	<	92	5Ch	\	124	7Ch		156	9Ch	Ĳ	188	BCh	ˆ	220	DCh	˜	252	FCh	ˆ
29	1Dh	GS (group separator)		61	3Dh	=	93	5Dh]	125	7Dh	}	157	9Dh	Ĳ	189	BDh	ˆ	221	DDh	˜	253	FDh	ˆ
30	1Eh	RS (record separator)		62	3Eh	>	94	5Eh	^	126	7Eh	~	158	9Eh	Ĳ	190	BEh	ˆ	222	DEh	˜	254	FEh	ˆ
31	1Fh	US (unit separator)		63	3Fh	?	95	5Fh	_				159	9Fh	Ĳ	191	BFh	ˆ	223	DFh	˜	255	FFh	ˆ
127	20h	DEL (delete)								theASCIIcode.com.ar														

D 肉肉搜尋器

TimeLimit: 1s

MemLimit: 1MB

題目敘述

據說有種東西叫做「肉肉搜尋器」，對喜歡吃肉肉的咖波而言這可是她夢寐以求的東西，就連睡覺時都會不斷想著它。

因為真的太想要它了，咖波上了一個有點非常奇怪的網站，清楚的標示肉肉搜尋器的運作原理。

『給你一個數列 $\langle a \rangle$ 。』

『你必須先生成一個數列 $\langle b \rangle$ ，定義為 $b[i] = a[i]$ 的各個位數相乘，若遇0則乘以10。』

『試求 $\langle b \rangle$ 中所有連續子序列中平均值最大的子序列。』

.....這網站絕對是在唬，所以咖波直接拉到最網頁的最底部。果然，這網站是在唬，要你先上傳一個可以AC掉上面那題目的扣才讓你去看肉肉搜尋器的運作原理。重！點！是！只要成功進入那個網頁甚至還有可能獲得一個肉肉搜尋器！這種機會咖波怎麼會放掉呢？

但是因為咖波的資訊真的很弱，都快要被當掉了，只好請你幫忙囉。

你總不會拒絕這麼可愛的貓貓蟲吧？

輸入說明

第一行會有一整數 n ，表示有 n 個元素。

接下來會有 n 個整數代表數列 $\langle a \rangle$ ，以空格隔開，表示 $a[0], a[1], a[2] \dots a[n-1]$

輸出說明

輸出 $\langle b \rangle$ 所有連續子序列中所能形成平均值最大的數值值，最後換行。

範例

Input	Output
6 12 18 20 16 19 20	20

提示

假設給你一個數列 $\langle a \rangle = \{87, 92, 16, 10\}$ ，則 $\langle b \rangle = \{8*7, 9*2, 1*6, 1*10\} = \{56, 18, 6, 10\}$
 子序列的意思是:在原數列中任取一些元素所形成的數列稱為子序列(可以全拿)
 所以連續子序列的意思是:在從原數列中取出要形成子序列的元素必須是連在一起的
 以上面的 $\langle b \rangle$ 為例， $\langle b \rangle$ 的連續子序列可以是 $\{56\}$ 、 $\{18, 6\}$ 、 $\{56, 18, 6\}$ 、 $\{56, 18, 6, 10\}$...還有很多，在此不全部列出來
 連續子序列的平均值則是將該數列中個元素相加後平均
 對於本題意思有不懂的歡迎提問

配分

額外限制	分數
$1 \leq n \leq 10$, $0 \leq a[i] \leq 10^2$	100
$1 \leq n \leq 10^4$, $0 \leq a[i] \leq 10^8$	30
$1 \leq n \leq 10^5$, $0 \leq a[i] \leq 10^{16}$	70

E 神奇的秘笈

TimeLimit: 1s

MemLimit: 512MB

題目敘述

Minson現在正忙著把昨天半夜CodeForces的 pE刻出來。據當事人表示：我睡覺睡到一半突然想到要怎麼debug，電，咖波不小心就跪下去了。突然回神的咖波趕快爬起來往下一個位子走去。

企鵝兄正在忙著修被玩壞的cms，螢幕上的終端機一直跳，咖波還是看不懂，又不小心跪下去了。

橘子君正在出裡社團事務，沒辦法，下個活動快要開始了，企劃書卻還沒趕出來。咖波沒有興趣，繼續往下個位子走去。

六一居士。嘎現在正旁著處理茶會的善後、學生會事務、少年什麼挖戈的東西忙著，一邊打文件一邊跟別人開會討論。咖波發現他在做的事情跟他弟沒兩樣，只是感覺比較厲害多了跟別人開會罷了。

下一個位子，「我姓潘」先生，現在正忙著出題目.....看著螢幕上顯示的費波什麼那契數列之類吧啦吧啦的，咖波也不懂，又轉頭走了。

隔壁的桐人大大（欠打啊），感覺很閒，據說他題目都出完了正在等別人pass給他題目讓他出題敘，正在爽爽打VG中。咖波不打GAME的，意興闌珊的爬走了。

許神正在算數學（還是化學？咖波是學渣，傻傻分不清楚）。

一不小心後面那排全部路過一遍了咖波突然發現好無聊，什麼事都做不了。突然間，他瞄到書櫃最下面那層有一本感覺很厲害的書。抽出來一看居然是傳世秘笈，聽說讀完之後就可以直接屌打IOI金牌（最好啦）了？！

書的第一頁寫著一道題目，照理說應該是最簡單的。

但是咖波卻看不懂，只好來求助路過的你，你正想要拒絕他時，他用淚眼汪汪的大眼看著你，你也不好意思拒絕他，所以接受了。

打了這麼多幹話，重點只有一句：

直接看下一欄輸入說明就好了。

.....拜託別來打我，我差不多該去避難了。

我還要去ION Camp啊啊啊啊啊啊！

輸入說明

這題目很簡單，簡言之就是有一個數列，是變形的費波那契數列，只取該數列的後一位。

接著會給你兩數 a, b ，求在 a 範圍內 b 出現的次數。

第一行有一數 n 代表有 n 筆測資，

接下來的 n 行每行皆有2數 a, b 。

測資保證 $a \leq 10^{15}$ && $0 \leq b < 10$

*請注意此數列沒有第0項，且第1項與第2項為1。

輸出說明

輸出 a 範圍內 b 出現的次數

範例

Input	Output
3	1
5 5	2
6 1	2
9 3	

配分

額外限制	分數
$a \leq 10$	60
$a \leq 10^{10}$	90
$a \leq 10^{15}$	150

F 箭頭

TimeLimit: 5s

MemLimit: 130MB

題目敘述

咖波在一中校園內亂晃，看到許 ~ ~ ~ 多箭頭，你也知道，咖波最喜歡箭頭了！
想要拍照但是卻沒帶相機QwQ，只好央求在一旁的導遊——tangerine1202——
幫忙記錄一下這些箭頭的形狀，然後回去要用記事本作畫出那張圖！！
據說這是一個大學長的技能，但是他今天沒有回來，只好求你幫忙了！

輸入說明

只有一個輸入 n 。 ($4 \leq n \leq 20$)

輸出說明

輸出一個箭頭，請參考範例。

範例

Input	Output
4	<pre> # *** ##### ***** ### ### </pre>
5	<pre> # *** ##### ***** ##### ##### ##### ##### </pre>

配分

額外限制	分數
學長很好心的給你兩個答案了w	300

G 想吃東西的咖波

TimeLimit: 0.5s

MemLimit: 100MB

題目敘述

正急著終於找到肉肉準備要咬下去的咖波，突然發現他只能接近到距離肉肉不到一公分的地方.....然後就被某種立場給擋住了！真是悲劇啊，明明都已經找到沒被吃掉的肉肉了，居然不能吃？！現在的咖波已經快要崩潰了，一崩潰的咖波會把你給吃掉！（不要懷疑，咖波在肚子餓時無所不吃。）

為了要拯救你自己的性命，你只好先安撫一下咖波的情緒，然後不段的搜尋線索，皇天不負苦心人，終於被你找到了，這是一個很聰明的鎖，你每提供一組數字，它會告訴你這數字比正解「大」還是「小」。

別以為那麼簡單～有誰說解出來就好了！咖波現在正餓得要命，不敢快解出來你就要被吃掉了！

因為咖波肚子很餓，所以他希望你試越少次越好，他會根據你嘗試的次數來給你分數喔！

作答說明

本題為互動題型，沒有輸入輸出，您可以下載本題的附件下來修改

實作說明

本題只提供C++，如果上傳C檔案，請自行承擔各種不可預期的結果

本題您必須實作一個`int Guess()`函式，此函式必須回傳一個整數`ans` ($0 < ans \leq 100000$)代表這個鎖的密碼，如果錯了將會得到0分

本題提供你一個函式`int Ask(int g)`你可以呼叫它來確認密碼，此函式的傳入值就是你猜的密碼，如果密碼是對的，回傳0，如果還要再大則回傳1，如果還要再小，則回傳-1

請在程式碼前面加上`#include "guess.h"`

本題上傳的code請不要有main函式

附件會提供您參考程式碼

你可以直接修改參考程式碼，本程式碼是可以直接編譯並執行的，執行時請輸入一個數字，代表密碼，它會告訴你，你的程式猜了幾次，請將參考程式碼與附件的`guess.h`放在同一個資料夾內

如有任何問題，歡迎詢問

評分說明

本題會根據你呼叫 `int Ask(int g)` 的次數來評分，評分函式如下：

假設你呼叫了 m 次

$$(-0.1172m^2 + 130) \% \quad \text{if}(m < 16)$$

$$(-0.53m^2 + 6.63m + 130) \% \quad \text{if}(m \geq 16 \ \&\& \ m \leq 20)$$

$$(-\log_2(m - 20) + 35) \% \quad \text{if}(m > 20)$$

以上公式代表你會獲得本題總分的多少百分比

你有可能獲得比總分更高的分數，而本題也會將多餘的分數列入，也就是說本題分數可能爆表

配分

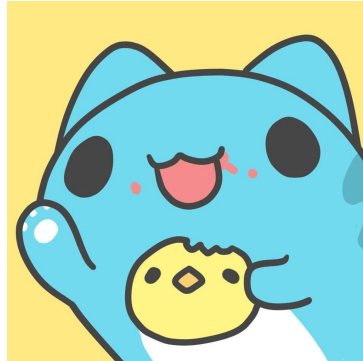
額外限制	分數
題目限制	100%

H 小雞們

TimeLimit: 2s

MemLimit: 100MB

題目敘述



「啾咕～」吃完肉肉的咖波，玩起他最愛玩的遊戲：「小雞農場」～只要玩過的人都知道，那遊戲非常舒壓，一群黃色的小雞在螢幕上跑來跑去，讓咖波非常喜歡（咖波最喜歡小雞了）。

玩著玩著，咖波突然發現這個遊戲可以切割成 $n * m$ 個單位，咖波很好奇在某些方格內的小雞數量，但是 n 與 m 實在太大了，只好再次請你幫忙囉～

他會給你兩個點，請將這兩個點圍起來的矩形內的小雞數量總和算出來

輸入說明

輸入第一行有三個整數 n, m, q ($0 \leq n, m \leq 1000$ $0 \leq q \leq 1000000$)

接下來有 n 行，每行 m 個整正數，代表每格的小雞數量 ($0 \leq \text{小雞數量} \leq 10^9$)

再來有 q 行，代表著 q 筆詢問，每行四個整數 x_1, y_1, x_2, y_2 ($0 \leq x_1, x_2 \leq m, 0 \leq y_1, y_2 \leq n$) 代表兩個點

左上角座標為 $(0,0)$ ，往右走 x 增加，往下走 y 增加

輸出說明

對於每筆詢問，請將兩個點圍起來的矩形內的小雞數量總和算出來

範例

Input	Output
3 4 2	78
1 2 3 4	34
5 6 7 8	
9 10 11 12	
0 0 3 2	
1 1 2 2	

配分

額外限制	分數
$n, m \leq 100, q \leq 1000$	250
題目範圍	150

I 肉肉在哪裡

TimeLimit: 3.9s

MemLimit: 256MB

題目敘述



「好餓 ~ ~ ~」中午12點了，一向吃飯很準時的咖波因為習慣性遲到的 Minson，遲遲不能去吃飯，心情也不好了起來（「Minson ~ ~ ~ 吃飯 ~ ~ ~」我絕對不會說是在影射某學長），於是便丟下T19的各位自己跑出去了！

只要認識咖波的都知道，咖波最愛吃肉肉了！為了方便尋找肉肉，他打開了一個走在科技尖端的產品：「肉肉搜尋器」！會發出一種人感覺不到、除了肉肉搜尋器以外的儀器收不到的電波，在遇到肉肉時反射回來，肉肉街搜尋器會用它內部不可告人的最高科技晶片運算肉肉在哪，並且將座標顯示在螢幕上！

但是咖波的空間概念不好（橘子：敢說咖波壞話，咖波，吃了他！），只知道有 n 快肉肉在平面上，所以只好再次跑回C302教室找T19的各位幫他的忙了，因為吃肉肉也有可能那塊肉已經先被別人吃掉了，咖波又不想要跑來跑去的，只好請T19的社員們幫他排序一下那些肉肉與自己的距離吧。

為了避免搞混，咖波還要求要同時輸出，那塊肉肉的編號喔！

輸入說明

第一行有三個數字（可能是整數，也可能是浮點數） x, y 以及整數 n 代表咖波現在所在的座標（ x, y ）以及共有 n 塊肉肉，第二行開始有 n 行，每行各有兩個數字， x_i, y_i 代表編號第 i 塊肉肉的位置。

測資保證 $0 \leq x, y \leq 10^6$ 以及 $n \leq 2.5 \times 10^6$

輸出說明

請由小到大輸出肉肉的編號及距離，如果有距離相同的肉肉，請以標號排序，格式請參考範例輸出

（喔對了，記得要輸出到小數點下三位）

範例

Input	Output
0 0 2 1 1 2 1	1: 1.414 2: 2.236

配分

額外限制	分數
範例測資	1
$n = 1, x, y \leq 1000$	5
$n = 2, x, y \leq 1000$	5
$x, y, n \leq 1000$, 且為整數	65
x, y 為整數	11
題目範圍	113
Master Points: $n = 2500000$	300

提示

輸出到小數點下兩位寫法：(先#include<iomanip>)

```
cout << fixed << setprecision ( [位數] ) << [內容] << endl;
```

J XOR

TimeLimit: 0.8s

MemLimit: 100MB

題目敘述

開心地做XOR吧!

C++做XOR的運算子是^，例如1 XOR 2，C++裡面可以寫成1 ^ 2

輸入說明

給定n,m代表有n個數字, m筆操作or詢問, ($1 \leq n \leq 10^5$, $1 \leq m \leq 10^5$), 接下來則有m行輸入:

1 l r x 代表將區間[l,r]的數字全部xor 數字x($1 \leq l \leq r \leq n$, $0 \leq x \leq 1000000000$)

2 p x 詢問位置x的數字在經過p次操作以後為多少,($0 \leq p \leq$ 已操作的次數)

輸出說明

一開始所有數字為0, 對於每個詢問輸出一行

範例

Input	Output
5 5	0
1 1 3 8	2
1 1 5 15	
2 0 3	
1 2 3 5	
2 3 2	

範例說明

位置3的數字經過0次操作答案為0(因為沒xor任何數字)

位置2的數字在經過3次操作之後為2 因為 $(8 \text{ xor } 15) \text{ xor } (5) = 2$

配分

額外限制	分數
$1 \leq n \leq 5000$ & $1 \leq m \leq 5000$	25%
$1 \leq n \leq 10^5$ & $1 \leq m \leq 10^5$	75%

K 大樹(hard)

TimeLimit: 1s

MemLimit: 100MB

題目敘述

給定一棵樹有 N 個點，請求出割出一個連通塊的最大邊權總合為多少

$$1 \leq N \leq 200000$$

輸入說明

給定一個 N 代表有 N 個節點的樹,接下來有 $N-1$ 行

$p \ w$

代表點 i 的父節點為 p ($2 \leq i \leq N$) 邊權為 w

輸出說明

輸出最大邊權總和

範例

Input	Output
5 1 -10 1 782 1 -453 1 -214	782

配分

額外限制	分數
$1 \leq N \leq 15$	20%
$1 \leq N \leq 200000$	80%

附錄1:

The ASCII code
American Standard Code for Information Interchange

www.theasciicode.com.ar

ASCII control characters			ASCII printable characters									Extended ASCII characters														
DEC	HEX	Simbolo ASCII	DEC	HEX	Simbolo	DEC	HEX	Simbolo	DEC	HEX	Simbolo	DEC	HEX	Simbolo	DEC	HEX	Simbolo	DEC	HEX	Simbolo	DEC	HEX	Simbolo	DEC	HEX	Simbolo
00	00h	NULL (carácter nulo)	32	20h	espacio	64	40h	@	96	60h	`	128	80h	Ç	160	A0h	à	192	C0h	À	224	E0h	Ó			
01	01h	SOH (inicio encabezado)	33	21h	!	65	41h	A	97	61h	a	129	81h	ü	161	A1h	á	193	C1h	Á	225	E1h	Ô			
02	02h	STX (inicio texto)	34	22h	"	66	42h	B	98	62h	b	130	82h	é	162	A2h	â	194	C2h	Â	226	E2h	Ö			
03	03h	ETX (fin de texto)	35	23h	#	67	43h	C	99	63h	c	131	83h	ê	163	A3h	ã	195	C3h	Ã	227	E3h	Ø			
04	04h	EOT (fin transmisión)	36	24h	\$	68	44h	D	100	64h	d	132	84h	ä	164	A4h	ä	196	C4h	Ä	228	E4h	ö			
05	05h	ENQ (enquiry)	37	25h	%	69	45h	E	101	65h	e	133	85h	å	165	A5h	å	197	C5h	Å	229	E5h	õ			
06	06h	ACK (acknowledgement)	38	26h	&	70	46h	F	102	66h	f	134	86h	ä	166	A6h	ä	198	C6h	Ä	230	E6h	μ			
07	07h	BEL (timbre)	39	27h	'	71	47h	G	103	67h	g	135	87h	ç	167	A7h	ç	199	C7h	Ç	231	E7h	þ			
08	08h	BS (retroceso)	40	28h	(72	48h	H	104	68h	h	136	88h	ø	168	A8h	ø	200	C8h	Ø	232	E8h	Û			
09	09h	HT (tab horizontal)	41	29h)	73	49h	I	105	69h	i	137	89h	é	169	A9h	é	201	C9h	É	233	E9h	Ü			
10	0Ah	LF (salto de línea)	42	2Ah	*	74	4Ah	J	106	6Ah	j	138	8Ah	è	170	AAh	è	202	CAh	È	234	EAh	Ý			
11	0Bh	VT (tab vertical)	43	2Bh	+	75	4Bh	K	107	6Bh	k	139	8Bh	ï	171	ABh	ï	203	CBh	Ï	235	EBh	Û			
12	0Ch	FF (form feed)	44	2Ch	,	76	4Ch	L	108	6Ch	l	140	8Ch	î	172	ACH	î	204	CAh	Î	236	EAh	Û			
13	0Dh	CR (retorno de carro)	45	2Dh	-	77	4Dh	M	109	6Dh	m	141	8Dh	ï	173	ADh	ï	205	CDh	Ï	237	EDh	Û			
14	0Eh	SO (shift Out)	46	2Eh	.	78	4Eh	N	110	6Eh	n	142	8Eh	ÿ	174	Afh	ÿ	206	CEh	ÿ	238	EEh	Û			
15	0Fh	SI (shift In)	47	2Fh	/	79	4Fh	O	111	6Fh	o	143	8Fh	ÿ	175	Afh	ÿ	207	CFh	ÿ	239	EFh	Û			
16	10h	DLE (data link escape)	48	30h	0	80	50h	P	112	70h	p	144	90h	ÿ	176	B0h	ÿ	208	D0h	ÿ	240	F0h	Û			
17	11h	DC1 (device control 1)	49	31h	1	81	51h	Q	113	71h	q	145	91h	ÿ	177	B1h	ÿ	209	D1h	ÿ	241	F1h	Û			
18	12h	DC2 (device control 2)	50	32h	2	82	52h	R	114	72h	r	146	92h	ÿ	178	B2h	ÿ	210	D2h	ÿ	242	F2h	Û			
19	13h	DC3 (device control 3)	51	33h	3	83	53h	S	115	73h	s	147	93h	ÿ	179	B3h	ÿ	211	D3h	ÿ	243	F3h	Û			
20	14h	DC4 (device control 4)	52	34h	4	84	54h	T	116	74h	t	148	94h	ÿ	180	B4h	ÿ	212	D4h	ÿ	244	F4h	Û			
21	15h	NAK (negative acknowle.)	53	35h	5	85	55h	U	117	75h	u	149	95h	ÿ	181	B5h	ÿ	213	D5h	ÿ	245	F5h	Û			
22	16h	SYN (synchronous idle)	54	36h	6	86	56h	V	118	76h	v	150	96h	ÿ	182	B6h	ÿ	214	D6h	ÿ	246	F6h	Û			
23	17h	ETB (end of trans. block)	55	37h	7	87	57h	W	119	77h	w	151	97h	ÿ	183	B7h	ÿ	215	D7h	ÿ	247	F7h	Û			
24	18h	CAN (cancel)	56	38h	8	88	58h	X	120	78h	x	152	98h	ÿ	184	B8h	ÿ	216	D8h	ÿ	248	F8h	Û			
25	19h	EM (end of medium)	57	39h	9	89	59h	Y	121	79h	y	153	99h	ÿ	185	B9h	ÿ	217	D9h	ÿ	249	F9h	Û			
26	1Ah	SUB (substitute)	58	3Ah	:	90	5Ah	Z	122	7Ah	z	154	9Ah	ÿ	186	BAh	ÿ	218	DAh	ÿ	250	FAh	Û			
27	1Bh	ESC (escape)	59	3Bh	;	91	5Bh	[123	7Bh	{	155	9Bh	ÿ	187	BBh	ÿ	219	DBh	ÿ	251	FBh	Û			
28	1Ch	FS (file separator)	60	3Ch	<	92	5Ch	\	124	7Ch		156	9Ch	ÿ	188	BCh	ÿ	220	DCh	ÿ	252	FCh	Û			
29	1Dh	GS (group separator)	61	3Dh	=	93	5Dh]	125	7Dh	}	157	9Dh	ÿ	189	BDh	ÿ	221	DDh	ÿ	253	FDh	Û			
30	1Eh	RS (record separator)	62	3Eh	>	94	5Eh	^	126	7Eh	~	158	9Eh	ÿ	190	BEh	ÿ	222	DEh	ÿ	254	FEh	Û			
31	1Fh	US (unit separator)	63	3Fh	?	95	5Fh	-				159	9Fh	ÿ	191	BFh	ÿ	223	DFh	ÿ	255	FFh	Û			

圖片來源:

<https://www.facebook.com/capoocat/>