



## Problem A Kamus Panda

Panda Buku, setelah membaca sekian banyak kamus (berbahasa Panda), memutuskan untuk menciptakan sebuah kamus Panda yang baru. Panda Buku mengajukan suatu cara pengurutan kata dalam kamusnya, yaitu berdasarkan *jenis karakter unik* yang ada pada kata. Walaupun rumit, cara ini dipakainya agar tidak dituduh menyontek kamus buatan Kutu Buku, yang terurut secara alfabetis.

Berikut ini adalah beberapa contoh cara menghitung *jenis karakter unik* dari sebuah kata yang dilakukan oleh Panda Buku:

- “lalala” terdiri dari 2 jenis karakter, yaitu ‘l’ dan ‘a’.
- “panda” terdiri dari 4 jenis karakter, yaitu ‘p’, ‘a’, ‘n’, dan ‘d’.

Adapun, dalam bahasa Panda, semua kata hanya terdiri dari huruf kecil dari ‘a’ sampai ‘z’. Semua karakter lain tidak dihitung sebagai bagian dari kata.

Karena Panda Buku sudah agak rabun dan kata yang dimilikinya amat banyak, anda diminta membuat program untuk menghitung *jenis karakter unik* untuk setiap kata yang diberikan oleh Panda Buku, agar ia dapat segera menyusun kamusnya.

### Input

Input diawali oleh satu baris dengan satu angka,  $T$  yang menandakan jumlah kata ( $T \leq 100$ ) yang dimiliki oleh Panda Buku. Sejumlah  $T$  kata menyusul di bawahnya, semuanya dalam baris tersendiri yang hanya berisi satu kata. Setiap kata terdiri dari maksimal 200 huruf.

### Output

Output harus terdiri dari tepat  $T$  baris, di mana tiap baris berisi satu angka, yaitu *jenis karakter unik* yang dimiliki setiap kata.

Contoh Input	Output untuk contoh input
3	2
lalala	4
panda	3
acm	

- halaman ini sengaja dikosongkan -



## Problem B Barisan Panda

Panda-panda Bu Ceemot akan berbaris di halaman. Setiap panda diberi nomor yang unik, yaitu dari  $1..N$  di mana  $N$  adalah jumlah pandanya. Para panda memiliki suatu Nilai Kesenangan terhadap suatu susunan barisan dari Bu Ceemot. Nilai kesenangan seekor panda adalah banyaknya panda di belakangnya, yang memiliki nomor lebih kecil. Nilai kesenangan para panda terhadap barisan ini adalah jumlah dari nilai kesenangan setiap panda.

Misalnya: para panda dibariskan dalam urutan 4 2 1 3. Artinya panda nomor 4 di urutan pertama, panda nomor 2 di urutan kedua, panda nomor 1 di urutan ketiga dan panda nomor 3 di urutan terakhir. Maka nilai kesenangan barisan ini dihitung sebagai berikut:

- Nilai kesenangan panda ke-1 adalah 3 (ada 3 panda di belakangnya dengan nilai lebih kecil, yaitu panda nomor 2, 1 dan 3)
- Nilai kesenangan panda ke-2 adalah 1 (ada 1 panda di belakangnya dengan nilai lebih kecil, yaitu panda nomor 1)
- Nilai kesenangan panda ke-3 adalah 0 (tidak ada panda di belakangnya dengan nilai lebih kecil)
- Nilai kesenangan panda ke-4 adalah 0 (tidak ada panda di belakangnya dengan nilai lebih kecil)

Maka total nilai kesenangan para panda terhadap barisan ini adalah  $3 + 1 + 0 + 0 = 4$ .

Sebelum membariskan pandanya, Bu Ceemot ingin mengetahui, ada berapa banyak barisan  $N$  panda dengan total nilai kesenangan para panda tepat  $K$ . Karena malas, ia meminta bantuan anda untuk menghitungnya. Bantulah Bu Ceemot menghitung angka ini. Karena nilainya bisa sangat besar, anda cukup menuliskan sisa baginya (modulus) dengan 1.000.007. Dua barisan dianggap berbeda jika dan hanya jika minimal ada satu panda yang menempati urutan yang berbeda.

### Input

Baris pertama berisi satu angka,  $T$  yang menandakan jumlah testcase ( $T \leq 100$ ). Setiap baris dari  $T$  baris di bawahnya menjelaskan satu testcase, dan terdiri dari dua bilangan bulat,  $N$  ( $1 \leq N \leq 200$ ) dan  $K$  ( $0 \leq K \leq N*(N-1)/2$ ).

### Output

Output terdiri dari  $T$  baris, dengan setiap baris berisi tepat satu bilangan bulat, yaitu jumlah kombinasi barisan  $N$  panda dengan total nilai kesenangan tepat  $K$ , setelah dimodulus dengan 1.000.007.

Contoh Input	Output untuk contoh input
4	1
3 0	2
3 1	2
3 2	1
3 3	



# BNPC

The 2008  
Bina Nusantara Programming Contest  
for High School Students



---

## Penjelasan contoh input dan output

Semua kemungkinan urutan barisan dengan 3 panda adalah:

- 1 2 3 -> Nilai kesenangan 0
- 1 3 2 -> Nilai kesenangan 1
- 2 1 3 -> Nilai kesenangan 1
- 2 3 1 -> Nilai kesenangan 2
- 3 1 2 -> Nilai kesenangan 2
- 3 2 1 -> Nilai kesenangan 3

Sehingga terdapat:

- 1 barisan dengan nilai kesenangan 0
- 2 barisan dengan nilai kesenangan 1
- 2 barisan dengan nilai kesenangan 2
- 1 barisan dengan nilai kesenangan 3



## Problem C

### Anti Rudal

Sejumlah rudal jarak jauh baru saja diluncurkan dan sedang mengarah ke tempat anda! Tidak ada waktu untuk mencari tahu siapa yang bertanggung jawab atas penyerangan ini. Sebagai komandan pasukan anti-rudal, anda harus segera mengaktifkan peluncur anti-rudal yang anda miliki dan mencegah bencana yang lebih buruk terjadi.

Peluncur anti-rudal bekerja dengan menembakkan anti-rudal (yang juga berupa rudal) dan meledakkan rudal sasarannya di udara. Sistem peluncur anti-rudal yang anda miliki ternyata memiliki kelemahan yang fatal. Jika anti-rudal sudah diluncurkan untuk meledakkan rudal pada ketinggian  $h$ , maka peluncur tersebut tidak bisa digunakan untuk meledakkan rudal pada ketinggian yang lebih rendah dari  $h$ . Setiap peluncur anti-rudal hanya bisa menembakkan satu anti-rudal dalam satu menit, dan satu anti-rudal hanya bisa meledakkan satu rudal sasaran.

Radarm yang anda miliki berhasil menangkap data-data yang diperlukan untuk meledakkan seluruh rudal musuh, yaitu:

- $t_{1..N}$ , menit tepat di mana anti-rudal harus diluncurkan untuk meledakkan rudal musuh ke- $i$
- $h_{1..N}$ , ketinggian rudal  $i$ . Asumsikan ketinggian rudal musuh tidak pernah berubah

Tidak ada dua rudal musuh yang memiliki  $t$  dan  $h$  yang sama.

Mengaktifkan sebuah peluncur anti-rudal ternyata membutuhkan bahan bakar dan biaya yang sangat besar. Meskipun ini situasi darurat, anda tetap harus memikirkan masa depan. Hitung berapa peluncur anti-rudal minimal yang perlu diaktifkan untuk menangkis semua serangan rudal musuh.

#### Input

Input baris pertama berisi sebuah bilangan bulat  $T$  ( $T \leq 100$ ), yang menyatakan jumlah kasus. Baris pertama pada setiap kasus berisi sebuah bilangan bulat  $N$  ( $1 \leq N \leq 100$ ) yang menyatakan jumlah rudal musuh. Baris kedua berisi  $N$  buah bilangan bulat  $t_{1..N}$  ( $1 \leq t_i \leq 100$ ) yang menyatakan menit tepat di mana anti-rudal harus diluncurkan untuk menangkis serangan rudal ke  $i$ . Baris ketiga berisi  $N$  buah bilangan bulat  $h_{1..N}$  ( $1 \leq h_i \leq 100$ ) yang menyatakan ketinggian rudal ke  $i$ .

#### Output

Cetak output untuk setiap kasus dalam satu baris yang berisi sebuah bilangan bulat yang menyatakan jumlah minimal peluncur anti-rudal yang perlu diaktifkan.

Contoh Input	Output untuk contoh input
1 6 2 4 1 5 3 2 5 6 2 7 3 6	3



# BNPC

The 2008  
Bina Nusantara Programming Contest  
for High School Students



---

## Penjelasan contoh input dan output

Diperlukan 3 peluncur anti-rudal untuk meledakkan semua rudal musuh,

- peluncur 1 : rudal #3 ( $t = 1, h = 2$ ), rudal #5 ( $t = 3, h = 3$ )
- peluncur 2 : rudal #6 ( $t = 2, h = 6$ ), rudal #4 ( $t = 5, h = 7$ )
- peluncur 3 : rudal #1 ( $t = 2, h = 5$ ), rudal #2 ( $t = 4, h = 6$ )



# BNPC

The 2008  
Bina Nusantara Programming Contest  
for High School Students



## Problem D

# Kelas Bermasalah

Pak Gaus kali ini sedang kesal dengan salah satu kelasnya, karena ternyata banyak muridnya yang tidak mengerjakan PR, bahkan ada yang sering membolos! Oleh karena itu ia ingin mengetahui seberapa banyak muridnya yang bermasalah, dengan hanya menganalisa daftar nilai yang dimilikinya.

Karena kesal, Pak Gaus tidak dapat berhitung dengan benar. Oleh karena itu, ia meminta bantuan anda untuk menghitung berapa murid di kelasnya yang memiliki nilai di bawah rata-rata. Adapun nilai rata-rata di sini dapat anda hitung dengan cara menjumlahkan nilai dari semua murid, kemudian dibagi dengan jumlah murid. Untuk lebih jelasnya, anda dapat memperhatikan contoh berikut ini.

Misalnya terdapat 4 orang murid, dengan nilai 25, 20, 30 dan 90 (terlihat kan, mana murid yang bermasalah dan mana yang tidak...). Rata-rata kelas dapat dihitung dengan mudah, yaitu  $(25+20+30+90) / 4 = 41,25$ . Oleh karena itu, dalam kasus ini, terdapat 3 orang murid yang nilainya di bawah rata-rata (yaitu murid yang nilainya 25, 20 dan 30).

### Input

Input diawali oleh satu baris dengan satu angka,  $T$  yang menandakan jumlah kasus ( $T \leq 100$ ). Penjelasan tiap kasus menyusul di bawahnya, dengan setiap kasus diwakili oleh dua baris. Baris pertama berisi sebuah angka  $N$ , yaitu jumlah murid di kelas ( $1 \leq N \leq 10.000$ ). Baris kedua berisi tepat  $N$  buah bilangan, yaitu nilai dari tiap murid. Perhatikan bahwa seperti biasanya, nilai minimal adalah 0 dan maksimal adalah 100.

### Output

Output harus terdiri dari tepat  $T$  baris, di mana tiap baris berisi satu angka, yaitu jumlah anak dengan nilai di bawah rata-rata kelas pada kasus yang dimaksud.

Contoh Input	Output untuk contoh input
2	3
4	1
25 20 30 90	
3	
45 50 55	

- halaman ini sengaja dikosongkan -





## Problem E

**Makan Sepuasnya**

Setelah memenangkan sebuah kompetisi pemrograman yang melelahkan, inilah saat yang tepat bagi para panda untuk berpesta di restoran masak-dan-makan-sepuasnya "Pandamasa". Di restoran ini makanan disediakan secara swalayan, di mana para pengunjung mengambil bahan makanan (mentah) untuk kemudian dimasak/dibakar di atas *plate* yang disediakan di meja sebelum disantap. Sesuai dengan naluri dan kebiasaan pada habitat alami mereka, para panda menggunakan "hukum rimba" yaitu mereka selalu memperebutkan makanan yang terdapat pada *plate* segera setelah makanan tersebut matang, tidak peduli siapa yang memasak makanan tersebut sebelumnya.

Selama pesta berlangsung, total ada  $N$  bahan makanan yang dimasak. Bahan makanan ke- $i$  dimasak pada waktu ke  $t_i$  (oleh salah satu panda) dan membutuhkan waktu selama  $d_i$  untuk matang. Setiap makanan yang sudah matang bisa dikonsumsi pada saat itu juga, dan waktu yang diperlukan untuk menikmati makanan tersebut adalah  $e_i$ . Setiap panda harus menghabiskan makanan yang sudah diambil sebelum diperbolehkan mengambil makanan yang lain (yang sudah matang) dari *plate*.

Pandut, salah satu panda yang mengikuti pesta ini, sangat lapar dan ingin makan sebanyak-banyaknya. Pandut memiliki kemampuan merebut makanan yang luar biasa, sehingga setiap kali ia ingin mengambil makanan dari *plate* maka ia pasti berhasil mendapatkannya. Tidak seperti teman-temannya yang pasti memperebutkan makanan setiap kali ada yang matang, Pandut mampu menahan diri untuk tidak ikut berebut jika total makanan yang bisa ia dapatkan bisa lebih banyak. Bantu Pandut untuk menghitung berapa jumlah maksimal makanan yang bisa ia dapatkan.

**Input**

Input diawali oleh satu baris dengan satu angka,  $T$  yang menandakan banyaknya skenario pesta yang diadakan. Input kemudian diikuti oleh  $T$  buah skenario yang masing-masing diawali oleh sebuah bilangan bulat  $N$  ( $1 \leq N \leq 100.000$ ) yang menandakan banyaknya makanan yang dimasak.  $N$  baris berikutnya masing-masing terdiri dari tiga bilangan bulat  $t_i$  ( $1 \leq t_i \leq 1.000.000$ ),  $d_i$  ( $1 \leq d_i \leq 1000$ ) dan  $e_i$  ( $1 \leq e_i \leq 1000$ ) yang menandakan waktu di mana makanan ke- $i$  mulai di masak, lama yang diperlukan untuk menjadi matang, dan lama yang diperlukan untuk mengkonsumsi makanan tersebut.

**Output**

Output harus terdiri dari tepat  $T$  baris, di mana tiap baris berisi satu angka, yaitu jumlah maksimal makanan yang bisa Pandut dapatkan.

Contoh Input	Output untuk contoh input
2	2
2	2
1 2 3	
2 4 2	
3	
0 2 3	
1 2 1	
2 2 1	

- halaman ini sengaja dikosongkan -



---

Problem F

## Fungsi Pak Gaus

Pak Gaus sedang mempersiapkan soal ulangan untuk muridnya. Topik ulangan kali ini adalah tentang operasi modulus. Tentu bukan Pak Gaus namanya kalau soal yang dipersiapkannya tidak unik, oleh karena itu, kali ini Pak Gaus telah mempersiapkan sebuah fungsi khusus. Fungsi tersebut definisinya adalah sebagai berikut.

$$f(a, b) = a^b \times (a - 1)^b \times (a - 2)^b \times \dots \times 1^b$$

Seperti yang telah diajarkan di sekolah dasar, kita tahu bahwa operasi pangkat adalah serangkaian perkalian, seperti yang ditunjukkan oleh contoh di bawah ini.

$$5^3 = 5 \times 5 \times 5$$

Oleh karena itu, fungsi Pak Gaus tersebut sebenarnya dapat dihitung dengan mudah, seperti contoh di bawah ini.

$$f(4,3) = (4 \times 4 \times 4) \times (3 \times 3 \times 3) \times (2 \times 2 \times 2) \times (1 \times 1 \times 1)$$

Dikarenakan hasilnya yang bisa besar sekali, dan karena ulangan ini sebenarnya untuk topik modulus, maka murid-murid diminta untuk menghitung hasil fungsi tersebut dan kemudian dimodulus dengan sebuah bilangan yang dipilih Pak Gaus.

Karena khawatir soalnya tidak dapat dikerjakan, Pak Gaus juga telah menyisipkan rumus berikut ini di soal ujian, agar murid-murid tidak frustrasi (mod adalah operator modulus).

$$(a \times b) \text{ mod } c = ((a \text{ mod } c) \times (b \text{ mod } c)) \text{ mod } c$$

Adapun operator modulus adalah operator sisa hasil bagi. Contoh:

- $5 \text{ mod } 3 = 2$
- $9 \text{ mod } 2 = 1$
- $7 \text{ mod } 4 = 3$

Pak Gaus tentu tidak kesulitan dalam membuat soalnya, karena yang dia perlu lakukan hanya memilih tiga angka ( $a$ ,  $b$ ,  $c$ ) secara acak, kemudian meminta murid-murid menghitung  $f(a, b) \text{ mod } c$ .

Namun tentu ia perlu membuat kunci jawabannya, dan anda diminta membantunya menghitung hasil dari perhitungan tersebut.



# BNPC

The 2008  
Bina Nusantara Programming Contest  
for High School Students



## Input

Input diawali oleh satu baris dengan satu angka,  $T$  yang menandakan jumlah soal yang dibuat Pak Gaus ( $T \leq 100$ ). Input kemudian diikuti oleh tepat  $T$  baris di mana tiap baris berisi tiga buah angka:  $a$ ,  $b$ , dan  $c$ , menjelaskan soal yang bersangkutan ( $1 \leq a, b \leq 10.000$ ;  $1 \leq c \leq 10.000$ ).

## Output

Output harus terdiri dari tepat  $T$  baris, di mana tiap baris berisi satu angka, yaitu hasil dari perhitungan sesuai dengan soal Pak Gaus di atas  $f(a,b) \bmod c$ .

Contoh Input	Output untuk contoh input
3	0
5 2 5	28
6 7 31	23
4 13 43	

## Penjelasan contoh input dan output

Input terdiri dari 3 kasus. Contoh perhitungannya untuk kasus ketiga:

$$\begin{aligned} f(4, 13) \bmod 43 &= (4 \times 4 \times \dots \times 4) \times (3 \times 3 \times \dots \times 3) \times (2 \times 2 \times \dots \times 2) \times (1 \times 1 \times \dots \times 1) \bmod 43 \\ &= 876488338465357824 \bmod 43 \\ &= 23 \end{aligned}$$



## Problem G

# Perubahan Minimal

Diberikan sebuah barisan dengan  $N$  bilangan acak (tidak harus unik), ubah barisan tersebut jika diperlukan dengan biaya sekecil mungkin sedemikian sehingga barisan yang baru memiliki semua bilangan dari 1 sampai  $N$  dengan urutan sembarang. Biaya untuk mengubah sebuah bilangan adalah sebesar selisih absolut bilangan tersebut sebelum dan sesudah perubahan (contoh: biaya untuk mengubah 10 menjadi 3 adalah 7).

Contoh. Barisan input terdiri dari 3 bilangan, yaitu 2, 4 dan 2. Karena urutan tidak dipermasalahkan, kita dapat memilih untuk mengubah barisan tersebut menjadi salah satu di antara barisan berikut:

- 1 2 3 (2 → 1, 4 → 2, 2 → 3) dengan total biaya:  $1 + 2 + 1 = 4$
- 1 3 2 (2 → 1, 4 → 3, 2 → 2) dengan total biaya:  $1 + 1 + 0 = 2$
- 2 1 3 (2 → 2, 4 → 1, 2 → 3) dengan total biaya:  $0 + 3 + 1 = 4$
- 2 3 1 (2 → 2, 4 → 3, 2 → 1) dengan total biaya:  $0 + 1 + 1 = 2$
- 3 1 2 (2 → 3, 4 → 1, 2 → 2) dengan total biaya:  $1 + 3 + 0 = 4$
- 3 2 1 (2 → 3, 4 → 2, 2 → 1) dengan total biaya:  $1 + 2 + 1 = 4$

Dengan demikian, biaya minimal yang diperlukan untuk mengubah  $\{2, 4, 2\}$  menjadi barisan yang memiliki semua bilangan dari 1 sampai 3 adalah 2, yaitu diubah menjadi  $\{1, 3, 2\}$  atau  $\{2, 3, 1\}$ .

### Input

Input diawali oleh satu baris dengan satu angka,  $T$  yang menandakan jumlah kasus ( $T \leq 100$ ). Penjelasan tiap kasus menyusul di bawahnya, dengan setiap kasus diwakili oleh dua baris. Baris pertama berisi sebuah angka  $N$  ( $1 \leq N \leq 10.000$ ), yaitu jumlah elemen pada barisan yang diberikan. Baris kedua berisi tepat  $N$  buah bilangan, yaitu barisan bilangan yang akan diubah (minimal 0, maksimal 100.000).

### Output

Output harus terdiri dari tepat  $T$  baris dengan tiap baris berisi satu angka, yaitu biaya minimal untuk mengubah barisan pada kasus yang dimaksud.

Contoh Input	Output untuk contoh input
3	2
3	1
2 4 2	0
4	
5 1 2 3	
2	
2 1	

### Penjelasan contoh input dan output

Untuk kasus pertama, kita dapat mengubahnya menjadi 1 3 2 atau 2 3 1 dengan biaya 2.

Untuk kasus kedua, kita dapat mengubahnya menjadi 4 1 2 3 dengan biaya 1.

Untuk kasus ketiga, kita tidak perlu melakukan perubahan, sehingga biayanya 0.

- halaman ini sengaja dikosongkan -



# BNPC

The 2008  
Bina Nusantara Programming Contest  
for High School Students



## Problem H

# Pasangan Paling "Wah"

Sebuah pesta dihadiri oleh tepat  $N$  anak laki-laki (dengan nomor urut 0 sampai dengan  $N-1$ ) dan  $N$  anak perempuan (dengan nomor urut 0 sampai dengan  $N-1$ ), yang semuanya masih belum mempunyai pasangan. Penyelenggara pesta menyadari bahwa pestanya akan menjadi lebih menyenangkan bila semua yang hadir dipasangkan. Artinya, setiap satu anak laki-laki akan dipasangkan dengan tepat satu anak perempuan, dan demikian pula sebaliknya.

Karena semua yang hadir malu untuk memilih pasangannya sendiri (dan untuk menghindari ada anak yang tidak mendapat pasangan), maka penyelenggara pesta memutuskan untuk menentukan sendiri cara memasangkannya. Agar tidak ada yang protes, penyelenggara pesta akan mengikuti sebuah algoritma untuk mendapatkan pasangan-pasangan paling "Wah" yang mungkin.

Setiap anak, baik laki-laki maupun perempuan akan diberikan sebuah nilai, sebut saja "faktor wah". Nilai "wah" dari sebuah pasangan adalah hasil kuadrat dari selisih "faktor wah" di antara keduanya. Karena pestanya akan segera dimulai, maka penentuan pasangan dilakukan dengan cara yang sederhana saja:

- (1) Dipilih satu anak laki-laki dan satu anak perempuan di antara yang belum mendapatkan pasangan, sedemikian sehingga nilai "wah" yang didapatkan setinggi mungkin. Apabila ada lebih dari satu pasangan dengan nilai "wah" yang sama, maka dipilih pasangan dengan nomor urut anak laki-laki terkecil. Apabila masih ada lebih dari satu pasangan, maka dipilih pasangan dengan nomor urut anak perempuan terkecil.
- (2) Ulangi langkah (1) hingga semua peserta pesta mendapatkan pasangan.

Karena jumlah peserta pesta cukup banyak, anda diminta untuk membuat program yang akan meng-generate "kombinasi pasangan – pasangan paling wah" dengan mengikuti metode di atas.

### Input

Input akan diawali dengan baris pertama berisi satu angka  $T$  ( $1 \leq T \leq 2500$ ), yang menyatakan jumlah skenario. Baris-baris berikutnya berisi detail skenario, dengan setiap skenario terdiri dari 3 baris:

- Baris pertama adalah  $N$  ( $1 \leq N \leq 10.000$ )
- Baris kedua berisi  $N$  buah bilangan, yaitu "faktor wah" dari anak laki-laki yang hadir sesuai urutan, dari 0 sampai  $N-1$
- Baris ketiga berisi  $N$  buah bilangan, yaitu "faktor wah" dari anak perempuan yang hadir sesuai urutan, dari 0 sampai  $N-1$

Untuk semua "faktor wah", nilai paling kecil adalah 0 dan paling besar adalah 10.000.

### Output

Untuk setiap skenario, cetak  $N$  pasangan dalam  $N$  baris berisi 3 angka: nomor urut anak laki-laki, nomor urut anak perempuan, dan nilai "wah" dari pasangan tersebut. Cetak pula sebuah baris kosong di akhir output untuk setiap skenario.



# BNPC

The 2008  
Bina Nusantara Programming Contest  
for High School Students



Contoh Input	Output untuk contoh input
3	0 0 0
1	
3	0 0 49
3	1 1 36
3	2 2 4
9 0 8	
2 6 6	6 6 81
10	8 4 81
1 1 3 5 8 3 0 6 9 2	0 9 64
7 7 2 8 0 3 9 2 4	1 3 49
	4 2 36
	9 0 25
	2 1 16
	7 7 16
	3 5 4
	5 8 1