

Департамент Информатика
Школа „Състезателно програмиране”
СЪСТЕЗАНИЕ, 11 май 2014 г.

Задача А. Красив номер

Някои хора дават луди пари за да получат “красив” номер за колата или телефона си. Станчо никога не е имал излишни пари и номерът на телефона му не е много “красив”. Това обаче не му пречи да размества цифрите в номера си и да се опитва да получи някоя красива комбинация. Критерият на Станчо е следният – той разбива номера на групи от 2, 3 или 4 последователни цифри и размества цифрите на групата по всевъзможни начини. На всеки тип комбинация от 2, 3 или 4 цифри Станчо е съпоставил “красивост” съгласно таблицата вдясно (където а, b и с са три различни цифри), а красотата на конкретното разбиване се определя от сумата от красотите на отделните групи. Красота на задения номер е красотата на най-доброто разместване. Ако комбинацията от цифри на групата я няма в таблицата, тогава красотата на групата е 0.

aa	2
aab	2
aaa	3
abc	2
aabb	3
aaab	4
aaaa	5

Например, красотата на разбиването 872-73-33 на номера 8727333 е $0+0+2=2$, а красотата на разбиването 8727-333 е $2+3=5$ (получена при разместването 7782 на цифрите на първата група). Напишете програма, която по зададен телефонен номер определя колко красив е този номер.

На първия ред на стандартния вход ще бъде зададен броят на тестовете. За всеки тест на един ред е зададен 7-цифрен телефонен номер.

За всеки тест, на отделен ред на стандартния изход, програмата трябва да изведе красотата на зададения на входа номер.

Пример**Вход**

2
8727333
8827291

Изход

5
4

Департамент Информатика
Школа „Състезателно програмиране”
СЪСТЕЗАНИЕ, 11 май 2014 г.

Задача В. Пътища

Маф живее в студентски град, в който има N дискотеки, които са номерирани с числата от 1 до N . Между някои двойки дискотеки (но не между всички) има директни пътища. В апартамента на Маф са ставали толкова купони, че спокойно можем да приемем, че той е една от дискотеките. Ако си представим, че дискотеките са върхове на граф, а директните пътища – ребра на графа, то този граф е кореново дърво, в който апартаментът на Маф е корен и има номер 1. Директният път между две дискотеки с номера X и Y е двупосочен и се преминава във всяка от посоките за време C_{xy} .

Маф е направила списък от двойки дискотеки (A, B) . Тя планира всеки ден да посети точно една двойка, като първо отиде от апартамента си в дискотека с номер A (първата дискотека от двойката), а от там, по-късно вечерта, се придвижи до дискотека B . Маф може да се движи само по директните пътища, които са ребра на дървото и, естествено, ако по един директен път мине два пъти, то времето за преминаването по него се удвоява. Къде ще ходи след дискотека B (и въобще ще ходи ли накъде) не е ясно, така че, за всяка вечер, Маф планира пътя си до дискотека B , минавайки първо през дискотека A . Възможно е да няма какво особено да се планира, тъй като пътят е единствен и за Маф остава само да пресметне колко време ще и отнеме придвижването от апартамента до дискотека A и от там до дискотека B .

Но...Маф има могъщ приятел ГГ (Голям Гръб), който е решил да минимизира усилията на възлюбената си, свързани с придвижването и от дискотека в дискотека. За целта, вечер той може да построява (без знанието на градските власти, разбира се) един нов (допълнителен към съществуващите), директен, двупосочен път, който се преминава за време T и свързва две различни дискотеки. Маф може да използва и този път при похода си по дискотеките. Целта е този път да се построи между такива две дискотеки, че да се минимизира времето за придвижване на Маф през съответната вечер. Както казахме, допълнителният път е „нелегален“ и, поради това, ГГ го разрушава до сутринта, а на следващата вечер прави нов - с ново време за преминаване и, евентуално, между други две дискотеки (но може и да са същите).

Напишете програма, която по зададен списък от двойки дискотеки, които Маф се кани да посещава и времената, за които ще се преминава по построените допълнителни пътища всяка вечер, пресмята минималните времена, които ще са необходими на Маф за придвижване между дискотеките всяка от планираните вечери.

Вход

От първия ред на стандартния вход се въвежда броя на тестовите примери. За всеки от тях: на първия ред е зададено едно цяло, положително число N – брой на дискотеките в студентски град (включително апартамента на Маф). От редове с номера $i=2,3,\dots,N$ се въвеждат по две цели числа P_i и C_i , разделени с интервал: P_i е номерът на дискотеката, която се явява „баща“ на дискотека номер i в дървото от дискотеки и преки пътища между тях, а C_i – времето, което е необходимо за придвижване между дискотеки с номера i и P_i .

От следващия ред се въвежда едно цяло, положително число Q – брой на вечерите, в които Маф ще посещава по две дискотеки. Следват Q реда, на всеки от които има по три цели числа A , B и T ($A \neq B$) – номерата на дискотеките, които Маф планира да посети (в този ред) в съответната вечер и времето, което ще е нужно за преминаване по допълнителния път, който ГГ ще построи през същата вечер.

Изход

За всеки тест на Q реда от стандартния изход изведете по едно число – минималното време, което ще е необходимо на Маф, за да посети планираните за съответната вечер дискотеки при подходящо построен допълнителен път.

Ограничения:

$$2 \leq N \leq 1\,000\,000, 1 \leq Q \leq 100\,000$$

$$0 \leq C_i \leq 1\,000$$

$$0 \leq T \leq 1\,000\,000$$

$$2 \leq A, B \leq N$$

Вход	Изход
1	8
4	17
1 4	
2 3	
2 5	
2	
3 4 1	
4 3 10	

Обяснение на примера

За първата вечер ГГ ще построи път между дискотеки 3 и 4 и след това Маф ще премине по маршрут $1 \rightarrow 2 \rightarrow 3 \rightarrow 4$.

За втората вечер, където и да построи ГГ допълнителния път, оптималният маршрут на Маф няма да го включва. Маф ще премине по маршрут $1 \rightarrow 2 \rightarrow 4 \rightarrow 2 \rightarrow 3$.

Департамент Информатика
Школа „Състезателно програмиране”
СЪСТЕЗАНИЕ, 11 май 2014 г.

Задача С. Рекорд

Разглеждаме низове, съставени от по 5 символа. Символите могат да бъдат само 0 и 1. Нека имаме множество от такива низове, подредени един под друг. За всяка двойка съседни низове преброяваме колко са единиците, заемащи едни и същи позиции и този брой наричаме **конфликт**. Например, ако 01101 и 01001 са един под друг, то те имат конфликт 2. Сумата от всички конфликти в така подреденото множество от низове наричаме **рекорд**. Нека са дадени пет неотрицателни цели числа b_k , $k = 1, 2, 3, 4, 5$. Напишете програма, която намира каква е минималната стойност на величината **рекорд** за множество от низове, което съдържа точно b_1 низа с по 1 единица, b_2 низа с по 2 единици и т.н.... b_5 низа с по 5 единици. Както се досещате величината рекорд зависи както от подреждането на единиците в низовете, така и от подреждането на низовете един под друг.

За всеки тест от отделен ред на стандартния вход се въвеждат 5 неотрицателни, цели числа, разделени с по един интервал – стойностите на b_k , $k = 1, 2, 3, 4, 5$.

За всеки тест на отделен ред на стандартния изход изведете едно цяло, неотрицателно число – намерената минимална стойност на величината рекорд.

Ограничение: $1 \leq b_1 + b_2 + b_3 + b_4 + b_5 \leq 10\,000$.

Вход	Изход
3 1 4 2 1	3

Обяснение на примера: Низовете избираме и подреждаме например така:

11110
00001
11100
00011
11100
10011
01110
00001
11110
00001
11111

В тази подредба има 3 низа с по 1 единица, 1 низ с 2 единици, 4 низа с по 3 единици, 2 низа с по 4 единици и 1 низ с 5 единици. Има само 3 съседни двойки низове, такива че във всяка двойка има по една единица в една и съща позиция.

Департамент Информатика
Школа „Състезателно програмиране”
СЪСТЕЗАНИЕ, 11 май 2014 г.

Задача D. Игра на карти

На всяка от n на брой карти е написано по едно цяло положително число, което е по-малко от 1000. Картите са подредени една до друга. Мартин е измислил нова игра. В нея се извършват $n - 2$ на брой хода. При всеки ход се премахва една от картите в редицата без първата и последната. За всяка отстранена карта се получават точки, които се изчисляват като се умножи числото, записано тази карта със сбора от числата, записани на съседните две. Накрая остават само две карти (първата и последната). Напишете програма, която намира най-големия брой точки, които могат да се получат при тази игра.

Вход

За всеки тест на първия ред на стандартния вход е зададено числото n ($3 \leq n \leq 700$). На следващия ред са зададени числата, написани върху картите, като е спазена подредбата им в редицата. Края на входа е маркиран с 0.

Изход

За всеки тест на стандартния изход да се изведе максималният брой точки, които могат да се получат при изиграване на описаната игра. Спазвайте изходния формат!

Вход	Изход
4 4 5 6 2 0	test case #1: 86

Департамент Информатика
Школа „Състезателно програмиране”
СЪСТЕЗАНИЕ, 11 май 2014 г.

Задача Е. Скокове

Разглеждаме редица от N цели числа. Стойността на първият елемент на редицата вземаме като начална стойност на наша текуща сума. Тръгваме от този начален елемент на редицата и се движим надясно чрез единични и двойни скокове. При всяко приземяване вземаме числото върху което стъпваме и го събираме с текущата сума, за да получим нова стойност на текущата сума. Понеже двойните скокове са поизморителни, не ни е разрешено да правим два двойни скока един след друг. Последният скок трябва да завърши върху последното число от редицата.

Напишете програма, която намира най-голямата стойност на текущата сума, която може да образуваме от взетите числа в някой от моментите на движението.

От първия ред на стандартния вход се въвежда едно цяло число N – броят на числата в редицата. От следващият ред се въвеждат N числа – числата на редицата. Края на входа е маркиран с нула.

На стандартния изход програмата трябва да изведе едно цяло число - търсената максимална сума.

Ограничения: $1 < N < 100\,000$; Числата в дадената редица имат стойности в диапазона от -99 до 99 (включително).

Вход	Изход
6 1 -3 -2 6 -3 -7 0	5

Пояснение: Резултат 5 се получава чрез движение, при което правим последователно един двоен и един единичен скок. Така събираме числата 1, -2 и 6. При всяка друга последователност от разрешени скокове ще имаме текуща сума, която не е по-голяма от 5 във всеки момент от началото до края на движението.

Департамент Информатика
Школа „Състезателно програмиране”
СЪСТЕЗАНИЕ, 11 май 2014 г.

Задача F. Вълшебен тоалет

Вики не е добра програмистка, но с вълшебствата си я бива. Днес тя е решила да си облече вълшебен тоалет – шалче, блуза, пола и обувки, които ѝ позволяват да бъде невидима когато си реши. Според книгата с вълшебства, вълшебен е този тоалет, при който разликата в цветовете на четирите му елемента е минимална.

В гардероба на Вики има N_1 шалчета, N_2 блузки, N_3 пола и N_4 чифта обувки. За всеки елемент от облеклото е известен неговия цвят (цяло число от 1 до 100 000). Всеки тоалет се характеризира с максималната разлика между кои да е два негови елемента и се състои от точно по едно шалче, блуза, пола и един чифт обувки.

Помогнете на Вики, като напишете програма, която намира елементите на вълшебния тоалет, т.е. тоалет при който максималната разлика в цветовете е минимална.

Вход

За всеки тест: За всеки тип облекло i ($i = 1, 2, 3, 4$) първо се въвежда броят N_i на елементите облекло от този тип, като на следващия ред се въвежда последователност от N_i цели числа, описващи цветовете на елементите. Четирите типа се въвеждат от стандартния вход последователно, започвайки от шалчетата и завършвайки с обувките. Край на входа е „end”.

Изход

За всеки тест на отделен ред на стандартния изход изведете четири цели числа – цветовете на шалчето, блузата, полата и обувките, които Вики трябва да избере, за да може да става невидима когато си поиска. Ако съществуват повече от един вълшебни тоалета, изведете кой да е от тях.

Ограничения: $1 \leq N_i \leq 100\,000$; Всички въвеждани числа са цели, положителни и не надвишават 100 000

Вход	Изход
5 3 8 5 8 10 3 5 2 4 4 9 7 7 3 5 10 8 3 10 9 end	3 4 3 3

Обяснение: В посоченият пример има точно два вълшебни тоалета: {3, 4, 3, 3} и {3, 2, 3, 3}. При тях максималната разлика между кои да е два цвята е минимална.



НОВ БЪЛГАРСКИ УНИВЕРСИТЕТ

София 1635, ул. Монтевидео 21
тел.: 55 81 37, 55 21 35, факс: 957 19 30

Департамент Информатика
Школа „Състезателно програмиране”
СЪСТЕЗАНИЕ, 11 май 2014 г.

Задача G. Тежки числа

Казваме, че числото a е по-тежко от числото b , ако сумата от цифрите на a е по-голяма от сумата на цифрите на b . Ако двете числа имат една и съща сума на цифрите, по-тежко е по-голямото от тях.

Напишете програма, която намира най-тежкото от три зададени числа.

Вход

На първия ред на стандартния вход е зададен броя на тестовите примери. За всеки пример се въвеждат трите цели числа – a , b и c .

Изход

За всеки тест на отделен ред на стандартния изход програмата трябва да изведе едно цяло число – най-тежкото от въведените три числа.

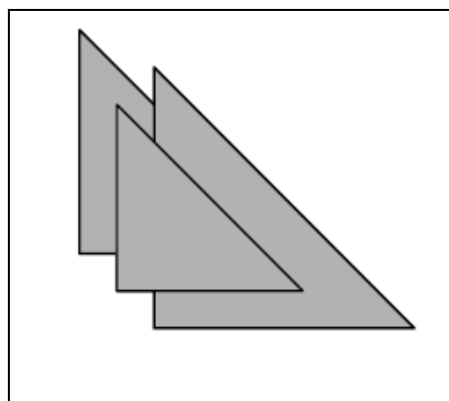
Ограничения: $9 < a < 100$; $99 < b < 1000$; $999 < c < 10000$

Вход	Изход
2	84
84 254 3125	264
84 264 3125	

Департамент Информатика
Школа „Състезателно програмиране”
СЪСТЕЗАНИЕ, 11 май 2014 г.

Задача Н. Ескадра

Ескадра чуждопланетни космически кораби се е приближила на опасно разстояние от важен за защитата на Земята обект. Когато ескадрата застане над обекта, защитниците му ще имат твърде кратко време (само един изстрел) за да я унищожат. Това обаче ще бъде възможно, само ако в бойния си полет космическите кораби са се разположили така, че проекциите им върху земната повърхност имат непразно сечение.



За радост на защитниците, всеки от корабите е с форма на правоъгълен равнобедрен триъгълник (носът на кораба е върхът при правия ъгъл) и при бойния си полет те застават така, че всички леви катети, както и всички десни катети са успоредни (виж Фигурата). Това силно увеличава шансовете на защитниците и все пак ще им е нужна вашата помощ. Напишете програма, която определя лицето на сечението на проекциите на всички кораби, защото това може да се окаже важно за решителния удар.

На първия ред на стандартния вход ще бъде зададен броят на тестовете. Всеки тест започва с ред, на който е зададен броят N на корабите ($1 \leq N \leq 100$). Следват N реда, на всеки от който, с три цели числа X , Y и S е зададено описанието на един от корабите. X и Y са координатите на носа на кораба в правоъгълна координатна система, с оси успоредни на катетите на корабите, а S е дължината на съответния катет ($-10^5 \leq X, Y \leq 10^5, 0 < S \leq 1000$).

За всеки тест, на отделен ред на стандартния изход, програмата трябва да изведе лицето на сечението на проекциите на корабите върху земната повърхност с три знака след десетичната точка.

Пример

Вход

```
1
3
2 4 6
4 2 7
3 3 5
```

Изход

```
4.500
```

Департамент Информатика
Школа „Състезателно програмиране”
СЪСТЕЗАНИЕ, 11 май 2014 г.

Задача I. Игра

Нашият прочут програмист Станчо обича компютърните игри и е замислил следната 3-D игра. Подземен лабиринт се състои от H нива с правоъгълна форма и размери $N \times M$. Всяко от нивата е разделено на $N \cdot M$ клетки с квадратна форма и съответните клетки на две съседни нива са точно една над друга. В някои от клетките не може да се влиза защото те представляват колони които поддържат горното ниво. В празните клетки може да се влиза само през стена от някоя от съседните празни клетки, като пробиването на стената отнема 5 секунди. От една клетка на по-горно ниво може да се влезе в съответната ѝ клетка на по-долното, като се пробие пода, което също отнема 5 секунди. Целта, разбира се е героя на играта, който се намира някъде в най-горното ниво да достигне по най-бързия начин в указана клетка на най-долното ниво, където го чака С-награда “с бронзов тен”. Напишете програма, която да определя минималното необходимо за целта време.

На първия ред на стандартния вход ще бъде зададен броят на тестовете. Всеки тест започва с ред, на който са зададени числата M , N и H ($2 \leq M, N, H \leq 50$). Следват H блока от по M реда всеки, с по N знака на ред, които описват поредните нива, започвайки от най-горното. Всяка празна клетка е представена с точка – ‘.’’, а всяка заета от колона – с малката латинската буква ‘o’. В най-горното ниво една от празните клетки е означена с ‘1’ и там се намира нашият герой, а в последното ниво една от празните клетки е означена с ‘2’ и там се намира наградата. Описанията на отделните нива са отделени едно от друго с празен ред.

За всеки тест, на отделен ред на стандартния изход, програмата трябва да изведе минималното необходимо време необходимо на героя да достигне до наградата, като съществуването на възможност да се достигне до наградата е гарантирано.

Вход

Изход

```
1
3 3 3
1..
oo.
...

ooo
..o
.o

ooo
o..
o.2
```

```
60
```