

## III ТУРНИР ЗА КУПАТА НА ДЕКАНА НА ФМИ НА СУ

26 Ноември, 2006 г.

### Задача А. Статистика

Както повечето от вас знаят, кандидат-асистентът Станчо има разнообразни интереси. В последно време той се е запалил по криптоанализа. Причината за това е, че колегата му Панчо се занимава много активно с криптография. Панчо много обича да изпраща кодирани любовни писма до Мара Бомбата – млада и красива докторантка в тяхната катедра. Тъй като Панчо е много уверен в алгоритмите за кодиране които използва, той изпраща съобщенията по публични канали, например чрез таблото на катедрата. Станчо е твърдо решен да разобличи Панчо и да го издаде на жена му. За целта трябва да декодира някое от съобщенията. Ако Станчо успее, Панчо ще трябва да излезе в "творчески отпуск", при което всичката минерална вода на катедрата остава Станчово владение.

Панчо всъщност използва много прост алгоритъм, който трансформира всяка буква поотделно в друга буква, чрез някаква биекция. Станчо добре знае, че такива алгоритми могат да бъдат разбити чрез статистически анализ. Поради тази причина, му е необходима програма **A**, която да намира броя на срещанията на дадена буква в даден низ. Както се сещате, тази програма трябва да бъде написана от вас.

#### Вход

На първия ред на стандартния вход ще бъде зададен броят  $T$  на тестовите примери, които програмата трябва да обработи. Всеки от следващите  $T$  реда съдържа низ от малки латински букви, следван от интервал и малка латинска буква.

#### Изход

За всеки от тестовите примери програмата трябва да изведе на стандартния изход по един ред с едно число – броят на срещанията на дадената буква в дадения низ.

#### Примерен вход:

```
3
milomilosredkov o
riverfishstone z
aaaab a
```

#### Примерен изход:

```
3
0
4
```

## III ТУРНИР ЗА КУПАТА НА ДЕКАНА НА ФМИ НА СУ

26 Ноември, 2006 г.

### Задача С. Телафиф

Известният GSM оператор Телафиф е замислил нова услуга – да организира база от данни с имената (низове с дължина до 64 знака, съставени от латински букви, цифри и знакът '\_' и започващи с буква) и телефонните номера (низове от максимум 32 цифри) на потребителите. По този начин, например, когато А се обади на В, на екрана на В излиза името на А, даже ако В няма А в телефонния си указател. Освен това, ще се избегне поддръжката на личен указател – добавянето на нови хора, сменянето на телефонни номера и т.н., което е доста досадно! Единственият проблем на Телафиф е, че нямат достатъчно добри програмисти и не знаят как да организират бързото отговаряне на многобройни въпроси от рода на „Кой е телефонният номер на XXXX...?“ или „На кого е телефонът с номер #####...?“. Напишете програма **С**, която да поддържа базата данни и да отговаря на въпроси колкото може по-бързо.

### Вход

Заявките се подават от стандартния вход по една на ред. Възможни са 5 вида заявки

INSERT <име> <телефонен#>	Добавя потребител с даденото име и телефонен номер в базата.
DELETE <име>	Изтрива информацията за потребителя с даденото име (името винаги ще съществува в базата).
LOOKUP <име_или_#>	Извежда информация за потребител. Ако аргументът е име, извежда телефонния номер. Ако е телефонен номер – извежда името.
LIST	Извежда на отделен ред за всеки потребител името му и телефонния му номер, разделени с един интервал и подредени в лексикографски ред на имената. Такива заявки ще са не повече от 100.
DIE	Програмата трябва да прекрати работата си.

### Изход

За всяка заявка LOOKUP програмата трябва да изведе ред с име или телефонен номер – в зависимост от заявката, а за всяка заявка LIST списъка на всички текущо регистрирани потребители, по един на ред. Форматът на всеки ред е „<име> <телефонен #>“ (без кавичките).

### Примерен вход:

```
INSERT Peter_Barth 01772773102
INSERT gibona85 03591464321
LOOKUP Peter_Barth
LIST
INSERT Bui_Phong 11345663
LOOKUP 03591464321
DELETE Peter_Barth
INSERT Peter_Barth 064102432
LIST
DELETE Bui_Phong
LOOKUP gibona85
DIE
```

### Примерен изход:

```
01772773102
Peter_Barth 01772773102
gibona85 03591464321
gibona85
Bui_Phong 11345663
Peter_Barth 064102432
gibona85 03591464321
03591464321
```

## III ТУРНИР ЗА КУПАТА НА ДЕКАНА НА ФМИ НА СУ

*26 Ноември, 2006 г.*

### Задача D. Игра със зарчета

Да разгледаме следната игра за двама. Тя се играе със стандартни 6-стенни зарчета, които в началото са хвърлени по произволен начин. Стандартното 6-стенно зарче представлява куб и на стените му са написани числата от 1 до 6, като сумата на числата от две противоположни стени е 7 (наистина, има две възможности да се направи такова зарче, но това не влияе на решението на задачата). Играчите се редуват да правят ходове. Един ход се състои в преобръщане на точно едно зарче. Под преобръщане разбираме, да поставим зарчето да легне на една от стените, която е съседна на стената на която лежи преди преобръщането. Валидни са само преобръщанията, които намаляват числото записано на горната стена на зарчето. Губи този от двамата играчи, който не може да направи ход, а другият играч печели. Напишете програма **D**, която по зададен брой на зарчетата намира броя на различните хвърляния, при които губи играчът, който е първи на ход. Зарчетата са различими и две хвърляния се различават, ако на поне едно зарче числото, което е отгоре е различно в двете хвърляния. Приемаме, че всеки играч играе оптимално, т.е. ако може да спечели, независимо от ходовете на противника, той го прави.

#### Вход

Входните данни се четат от стандартния вход. Програмата трябва да може да обработи няколко тестови примера, като за всеки тестов пример на отделен ред ще бъде зададено едно число в интервала [1..64] – броят на зарците. Входните данни завършват с края на файла, няма да има празни редове между отделните тестови примери, както и преди първия и след последния тестов пример.

#### Изход

За всеки тестов пример програмата трябва да изведе на отделен ред едно число – броя на случаите, в които първият играч губи.

#### Примерен вход:

1  
2  
3

#### Примерен изход:

2  
12  
56

## III ТУРНИР ЗА КУПАТА НА ДЕКАНА НА ФМИ НА СУ

26 Ноември, 2006 г.

### Задача Е. Галактически съюз

В навечерието на приемането на Земята за член на Галактическият Съюз (ГС), комисарите по приемането, марсианците М'пешо и М'гошо, не са удовлетворени от напредъка, който е показала Земята. Макар всички документи да са готови и няма юридически причини за блокиране на членството, те смятат че нашата планета все още има проблеми с инфраструктурата на транспорта. Тъй като най-големите проблеми в тази сфера са от централната държава на Земята – България, те молят да им помогнете.

Марсианците разполагат с двумерна пътна карта на България, зададена в Декартова координатна система, в която всички пътища са представени като отсечки. Понеже най-много катастрофи стават по кръстовищата, М'пешо и М'гошо предлагат на всяко кръстовище да се построи денонощно заведение от веригата за бързо хранене, от която те имат акции. В заведението да се продава качествена бира и да се предават футболни мачове от Галактическата Лига. Тъй като интересът към Лигата в България е голям, вместо да катастрофират, хората ще паркират по кръстовищата, ще гледат мачове и ще пийват бира.

За целта трябва да се построи телевизионна кула в някаква точка  $(X,Y)$ , която да осигурява препращането на сигнал от Марс към заведенията. Марсианските технологии са много усложнени, затова единственото което знаем за тях е, че цената за предаването на сигнал от кулата до заведението с координати  $(A,B)$  е равна на евклидовото разстояние между тях на квадрат, т.е.  $(X - A)^2 + (Y - B)^2$  галона (*галон* е валутата на ГС). Разбира се, от ГС отпускат минимален бюджет за проекта, затова трябва общата сума за предаване на сигнала до всички набелязани кръстовища да е минимална. Напишете програма **Е**, която решава проблема.

#### Вход

Всеки тестов пример се чете от стандартния вход и започва с ред, съдържащ числото  $M$  ( $2 \leq M \leq 32$ ), задаващо броя на пътищата, които интересуват М'пешо и М'гошо. Следват  $M$  реда, всеки от които съдържа по четири, разделени с интервали, цели числа  $X_1, Y_1, X_2, Y_2$ , обозначаващи двата края на един от пътищата. Ако два пътя се пресичат, пресечната им точка е кръстовище, ако не съвпада с някой от краищата на двата пътя. Винаги ще има поне две различни кръстовища, образувани от зададените пътища и никои три от пътищата няма да се пресичат в едно и също кръстовище. На последния ред на стандартния вход ще е зададено числото 0.

#### Изход

За всеки тестов пример програмата ви трябва да изведе по един ред на стандартния изход, съдържащ две числа, разделени с интервал – избраните от програмата координати  $(X,Y)$  на телевизионната кула. Тъй като парите са на ГС и има строг финансов контрол, то за максимална прецизност, всяка координата трябва да бъде изведена във вида  $P/Q$ , където  $P$  и  $Q$  са цели взаимнопрости числа и  $Q$  е положително. В случай, че  $P$  е 0, то  $Q$  трябва да бъде 1.

#### Примерен вход:

```
6
-2 -2 2 2
-2 2 2 -2
-1 3 1 5
1 3 -1 5
3 -1 5 1
5 -1 3 1
0
```

#### Примерен изход:

```
4/3 4/3
```

## III ТУРНИР ЗА КУПАТА НА ДЕКАНА НА ФМИ НА СУ

*26 Ноември, 2006 г.*

### Задача F. Изгаряне

Мрежа с  $N$  възела, означени с  $1, 2, \dots, N$ , и  $M$  нишки е изплетена така, че всеки от двата края на нишка е вплетен в някой от възлите на мрежата заедно с краища на (една или повече) от останалите нишки. Мрежата е цяла, т.е. възможно е да бъде хваната за някой от възлите и да бъде повдигната. Всички нишки са произведени от горящ с огромна скорост материал. Когато бъдат запалени, те изгарят практически мигновено. Възлите от своя страна изгарят по-бавно. Можем да приемем, че всеки възел изгаря за 1 секунда. Една нишка започва да гори в някой от двата си края, когато възелът в който е вплетен край изгори напълно. Един възел започва да гори, когато някоя от нишките вплетени в него е запалена от другия край. Напишете програма **F**, която да намира времето, което ще измине от запалването на някой от върховете до момента, в който мрежата ще престане да бъде цяла. Мрежата не е цяла, ако е съставена от поне две части, всяка от които има поне един неизгорял възел.

#### Вход

На първия ред на стандартния вход ще бъде зададен броят на тестовите примери. Всеки тестов пример започва с ред, на който, разделени с един интервал, ще бъдат зададени целите положителни  $N$  и  $M$  ( $3 \leq N \leq 100$ ,  $3 \leq M \leq 5000$ ). На всеки от следващите  $M$  реда ще бъдат зададени по две цели числа  $I$  и  $J$ , разделени с интервал, които описват по една нишка.  $I$  и  $J$  са двата възела, в които са вплетени краищата на нишката. На последния ред е зададен възелът  $A$ , в който ще бъде запалена мрежата.

#### Изход

За всеки от тестовите примери програмата трябва да изведе на отделен ред на стандартния изход намереното време. Ако мрежата изгаря без да загуби целостта си, тогава програмата трябва да изведе  $-1$  за съответния тестов пример.

#### Примерен вход:

```
2
5 5
1 2
2 4
4 3
4 5
1 5
2
3 3
1 2
1 3
2 3
1
```

#### Примерен изход:

```
2
-1
```