GLOBUSSOFT

TECHNOLOGY AHEAD OF TIME

Greetings From Globussoft

- Given below are 5 Programming questions, you have to solve any 3 out of 5 questions.

- These 5 questions you can attempt in any technology like C/C++, java, .Net, PHP

- To solve these 3 questions you've max. 3 hours.

- While Solving these questions you are not allowed to use any **Search Engine** like Google, Yahoo, Bing …

All the best for your test

Globussoft

QUESTION – 1

In poker, you have 5 cards. There are 10 kinds of poker hands (from highest to lowest):

- royal flush - ace, king, queen, jack and ten, all in the same suit
- straight flush - five cards of the same suit in sequence, such as 10,9,8,7,6 of clubs; ace can be counted both as the highest card or as the lowest card - A,2,3,4,5 of hearts is a straight flush. But 4,3,2,A,K of hearts is not a straight flush - it's just a flush.
- four of a kind - four cards of the same rank, such as four kings.
- full house - three cards of one rank plus two cards of another rank
- flush - five cards of the same suit (but not a straight flush)
- straight - five cards in order - just like the straight flush, but mixed suits
- three of a kind - three cards of one rank and two other cards
- two pairs - two cards of one rank, two cards of another rank, and one more card
- pair - two cards of the same rank
- high card - none of the above

Write a program that will help you play poker by telling you what kind of hand you have.

Input

The first line of input contains the number of test cases (no more than 20). Each test case consists of one line - five space separated cards. Each card is represented by a two-letter (or digit) word. The first character is the rank (A,K,Q,J,T,9,8,7,6,5,4,3 or 2), the second character is the suit (S,H,D,C standing for spades, hearts, diamonds and clubs). The cards can be in any order (but they will not repeat).

Output

For each test case output one line describing the type of a hand, exactly like in the list above.

Example

Input:

```
3
AH KH QH TH JH
KH 5S 3C 5C 7D
QH QD 2S QC 2C
```

Output:

```
royal flush
pair
full house
```

QUESTION – 2

Young people spend a lot of money on things like sweets, music CDs, mobile phones and so on. But most young girls/boys have one problem: Their pocket money is not enough for all these jolly things. Little Lisa Listig is one of these poor girls with a small pocket money budget. Last month her pocket money lasted only one week. So she decided to enter into negotiations with her father. Her father Tomm - a mathematician - had an incredibly ingenious idea: He wrote down some fancy digits with operators (+,*) in between them on a sheet of paper and allowed Lisa to insert brackets. Then he declared that the result of that arithmetic expression is Lisa's new pocket money. Now it's Lisa's task to maximize her pocket money. As her father was surprised what a huge sum of money Lisa got for her result, he decided to minimize the result of the expression for his son Manfred. Now it's your task to calculate the results obtained by Lisa and her father.

Input

The first line of input contains the number of testcases k (k< 5000). Each of the following k lines consists of an arithemtic expression. This expression consists of numbers (0-9) seperated by one of the two operators '*' and '+'. There are no spaces between the characters. Each line contains less than 100 characters.

Output

For each expression output the result obtained by Lisa and the result obtained by her father separated by one space. The results of the calculations are smaller than 2^{64}.

Example

Input:

```
1
1+2*3+4*5
```

Output:

```
105 27
```

QUESTION – 3

The municipal chronicals of an unbelievable lordly major town in a land far, far away tell the following story:

Once upon a time the new crowned king Günther decided to visit all towns in his kingdom. The people of the unbelievable lordly major town expected that king Günther would like to see some of the most famous buildings in their town. For the lordly citizens it seemed neccessary that all streets in the town that the king would have to use had to be cobbled with stone. Unfortunately the unbelievable lordly major town had not much money at that time as they used most of their savings to erect the highest cathedral the world had ever seen.

Roumours were afloat that the real reason for their thriftiness was not that the town treasury was

empty but that many people believed that king Günther came to the throne by deceiving his father king Erwin and that in his youth he made a pact with the devil. But anyway, the citizens of the unbelievable lordly major town decided to pave only as much streets as were absolutely necessary to reach every major building.
Can you help the citizens of the unbelievable lordly major town to find out which streets should be paved?
It might be usefull to know that all major buildings are either at the end of a street or at an intersection. In addition to that you can assume that all buildings are connected by the given streets.

Input

t [number of testcases (1 <= t <= 100)]
p [price to pave one furlong of street (positive integer)]
n [number of main buildings in the town (1 <= n <= 1000)]
m [number of streets in the town (1 <= m <= 300000)]
a b c [street from building a to building b with length c (lengths are given in furlong and the buildings are numbered from 1 to n)]

Output

For each testcase output the price of the cheapest possibility to reach all main buildings in the city on paved streets. You can assume that the result will be smaller than 2^32.

Example

Input:

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

Output:

```
12
```

QUESTION – 4

There are n boxes on the circle. The boxes are numbered from 1 to n (1<=n<=1000) in clock wise order. There are balls in the boxes, and the number of all the balls in the boxes is not greater than n.

The balls should be displaced in such a way that in each box there remains no more than one ball. In one move we can shift a ball from one box to one of it's neighboring boxes.

Write a program that: reads from the standard input the number of boxes n and the arrangement of balls in the boxes, computes the minimal number of moves necessary to displace the balls in such a way that in each box there remains no more than one ball, writes the result in the standard output.

Input

The first line of the input file contains an integer t representing the number of test cases (t<=20). Then t test cases follows. Each test case has the following form:

- The first line contains one positive integer n - the number of boxes
- The second line contains n nonnegative integer separated by single spaces. The i-th number is the number of balls in the i-th box.

Output

For each test case, output one nonnegative integer - the number of moves necessary to displace the balls in such a way that in each box there remains no more than one ball.

Example

```
Input:
1
12
0 0 2 4 3 1 0 0 0 0 0 1

Output:
19
```

QUESTION – 5

Yesterday was Sam's birthday. The most interesting gift was definitely the chessboard. Sam quickly learned the rules of chess and defeated his father, all his friends, his little sister, and now no one wants to play with him any more.

So he decided to play with another birthday gift – a Book of Math Problems for Young Mathematicians. He opened the book somewhere in the middle and read the following problem:

"How many knights can be placed on a chessboard without threatening each other?" After a while he realized that this was trivial and moved on to the next problem: "How many bishops can be placed on a chessboard without threatening each other?". Sam is in trouble here. He is not able to solve this problem and needs your help.

Sam's chessboard has size N *x* N. A bishop can move to any distance in any of the four diagonal directions. A bishop threatens another bishop if it can move to the other bishop's position. Your task is to compute the maximum number of bishops that can be placed on a chessboard in such a way that no two bishops threaten each other.

Input

The input file consists of several lines. The line number i contains a single number N representing the size of the i-th chessboard. [N <= 10^100]

Output

The output file should contain the same number of lines as the input file. The i-th line should contain one number – the maximum number of bishops that can be placed on i-th chessboard without threatening each other.

Example

```
Input:
2
3

Output:
2
4
```