Problem name: CANDY

Jennifer is a teacher in the first year of a primary school. She has gone for a trip with her class today. She has taken a packet of candies for each child. Unfortunately, the sizes of the packets are not the same.

Jennifer is afraid that each child will want to have the biggest packet of candies and this will lead to quarrels or even fights among children. She wants to avoid this. Therefore, she has decided to open all the packets, count the candies in each packet and move some candies from bigger packets to smaller ones so that each packet will contain the same number of candies. The question is how many candies she has to move.

Input specification

The input file consists of several blocks of data. Each block starts with the number of candy packets **N**(1<= **N** <=10000) followed by **N** integers (each less than 1000) in separate lines, giving the number of candies in each packet. After the last block of data there is the number -1.

Output specification

The output file should contain one line with the smallest number of moves for each block of data. One move consists of taking one candy from a packet and putting it into another one. If it is not possible to have the same number of candies in each packet, output the number -1.

Example

Input file:

```
5
1
1
1
1
6
2
3
4
-1
```

Output file:

```
4
-1
```

Time limit: 1s
Source limit:50000B

Problem name: THRBL

Bob has unusual problem. In Byteland we can find a lot of hills and cities. King of Byteland ordered Bob to deliver magic balls from one city to another. Unfortunately, Bob has to deliver many magic balls, so walking with them would take too much time for him. Bob came up great idea - catapulting them.

Byteland is divided into intervals. Each interval contains city and hill.

Bob can catapult magic ball accurately from city A to city B, if between them there isn't higher hill than A's hill.

Input

Every test case contains N and M (N<=50000) (M<=50000), number of intervals and number of balls.

In next line there's N numbers H(H<=10^9) separated by one space.

In next M lines numbers A and B (1<=A,B<=N), number of city from which we want to catapult the ball and number of city to which we want to catapult the ball.

Output

Write one number - number of magic balls that Bob can catapult successfully.

Example

Input:

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

Output:

```
2
```

```
Bob can catapult ball number 1 and 3.
```

Time limit: 1s
Source limit:50000B

Problem name: COINS

In Byteland they have a very strange monetary system.

Each Bytelandian gold coin has an integer number written on it. A coin n can be exchanged in a bank into three coins: n/2, n/3 and n/4. But these numbers are all rounded down (the banks have to make a profit).

You can also sell Bytelandian coins for American dollars. The exchange rate is 1:1. But you can not buy Bytelandian coins.

You have one gold coin. What is the maximum amount of American dollars you can get for it?

Input

The input will contain several test cases (not more than 10). Each testcase is a single line with a number n, 0 <= n <= 1 000 000 000. It is the number written on your coin.

Output

For each test case output a single line, containing the maximum amount of American dollars you can make.

Example

`Input:`
```
12
2
```

`Output:`
```
13
2
```

You can change 12 into 6, 4 and 3, and then change these into \$6+\$4+\$3 = \$13. If you try changing the coin 2 into 3 smaller coins, you will get 1, 0 and 0, and later you can get no more than \$1 out of them. It is better just to change the 2 coin directly into \$2.

Time limit: 9s
Source limit:50000B

Problem name: AGGRCOW

Farmer John has built a new long barn, with N (2 <= N <= 100,000) stalls. The stalls are located along a straight line at positions x1,...,xN (0 <= xi <= 1,000,000,000).

His C (2 <= C <= N) cows don't like this barn layout and become aggressive towards each other once put into a stall. To prevent the cows from hurting each other, FJ want to assign the cows to the stalls, such that the minimum distance between any two of them is as large as possible. What is the largest minimum distance?

Input

t – the number of test cases, then *t* test cases follows.
* Line 1: Two space-separated integers: N and C
* Lines 2..N+1: Line i+1 contains an integer stall location, xi

Output

For each test case output one integer: the largest minimum distance.

Example

Input:

```
1
5 3
1
2
8
4
9
```

Output:

```
3
```

Output details:

FJ can put his 3 cows in the stalls at positions 1, 4 and 8,
resulting in a minimum distance of 3.

Time limit: 2s
Source limit:10000B

Problem name: BASE

The Really Neato Calculator Company, Inc. has recently hired your team to help design their Super Neato Model I calculator. As a computer scientist you suggested to the company that it would be neato if this new calculator could convert among number bases. The company thought this was a stupendous idea and has asked your team to come up with the prototype program for doing base conversion. The project manager of the Super Neato Model I calculator has informed you that the calculator will have the following neato features:

- It will have a 7-digital display.
- Its buttons will include the capital letters A through F in addition to the digits 0 through 9.
- It will support bases 2 through 16.

The input for your prototype program will consist of one base conversion per line. There will be three numbers per line. The first number will be the number in the base you are converting from. The second number is the base you are converting from. The third number is the base you are converting to. There will be one or more blanks surrounding (on either side of) the numbers. There are several lines of input and your program should continue to read until the end of file is reached.

The output will only be the converted number as it would appear on the display of the calculator. The number should be right justified in the 7-digit display. If the number is to large to appear on the display, then print ERROR" (without the quotes) right justified in the display.

A sample input file is shown here:

```
1111000  2 10
1111000  2 16
2102101  3 10
2102101  3 15
  12312  4  2
     1A 15  2
1234567 10 16
   ABCD 16 15
```

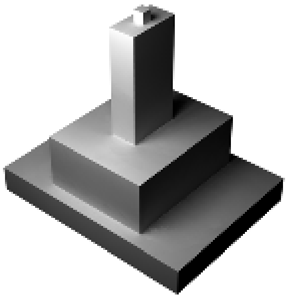
The following output file should be produced from the above sample input:

```
    120
     78
   1765
    7CA
  ERROR
  11001
 12D687
   D071
```

Time limit: 1s
Source limit:50000B

Problem name: BABTWR

Apart from the Hanging Gardens the Babylonians (around 3000-539 b.c.) built the Tower of Babylon as well. The tower was meant to reach the sky, but the project failed because of a confusion of language imposed from much higher above.



For the 2638th anniversary a model of the tower will be rebuilt. *n* different types of blocks are available. Each one of them may be duplicated as many times as you like. Each type has a height *y*, a width *x* and a depth *z*. The blocks are to be stacked one upon eachother so that the resulting tower is as high as possible. Of course the blocks can be rotated as desired before stacking. However for reasons of stability a block can only be stacked upon another if *both* of its baselines are shorter.

Input

The number of types of blocks *n* is located in the first line of each test case. On the subsequent *n* lines the height y_i, the width x_i and the depth z_i of each type of blocks are given. There are never more than 30 different types available.

There are many test cases, which come one by one. Input terminates with n = 0.

Output

For each test case your program should output one line with the height of the highest possible tower.

Example

Sample input:

```
5
31 41 59
26 53 58
97 93 23
84 62 64
33 83 27
1
1 1 1
0
```

Sample output:

```
342
1
```

Time limit: 3s
Source limit:50000B

Problem name: WORD

Ivana made up a long word of N letters. Then she wrote down all K-letter-substrings of that word. For example, if the original word is BANANA and K=3, Ivana writes down the words BAN, ANA, NAN, ANA. The number of these words is, obviously, N-K+1.

Ivana sorted these words in lexicographic order (in the given example, that would be ANA, ANA, BAN, NAN).

But the sad thing happened: Ivana forgot the original word! Your task is to reconstruct it. A unique solution will exist in all of the test data.

Constraints: $3 \leq N \leq 100\ 000$, $2 \leq K \leq 15$, $K < N$.

Input

[integers N, K]

[N-K+1 words in lexicographic order, each consisting of capital English letters]

Output

[the required word]

Example

Input:

```
6 3
ANA
ANA
BAN
NAN
```

Output:

```
BANANA
```

Time limit: 3s
Source limit:50000B