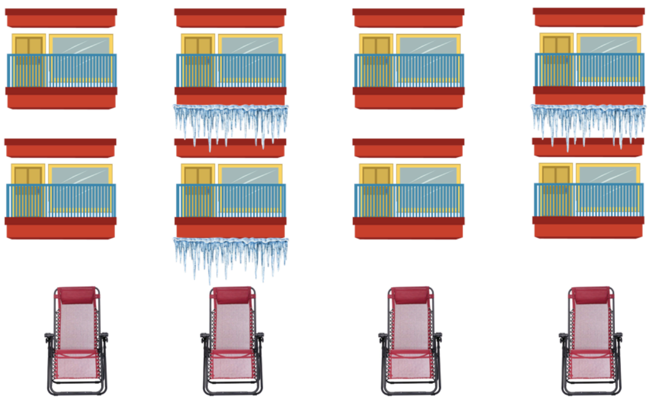
2020 UCC Coding Competition
Problems and Test Data by Kevin Yunqiao Liu

Problem 1: Snowstorm

You are the manager of a luxury hotel's guest pool and spa. You want to make sure each guest visiting the pool and spa area can have their own lounge chairs by the poolside to relax in. Your pool is outdoors, and the N lounge chairs are placed in a line by the hotel building. Two rows of balconies from the hotel's second and third floors hang over each lounge chair, as shown in the diagram below:


Images from FreePNG, Amazon

A huge snowstorm just hit your hotel, covering some of the balconies with snow and icicles. You have decided that it is not safe to allow a guest to use a lounge chair if either of the two balconies hanging over it is covered in snow and ice. For example, in the diagram above, the first and third lounge chairs are safe for use but the second and fourth are not.

Please determine how many lounge chairs are safe for use, so you know how many guests you can let into the pool at a time.

Input Specification

The input will consist of three lines. The first line contains N, the number of lounge chairs you have in a line.

The second line describes the snow and ice conditions on the first row of balconies above the lounge chairs. It contains a string of N 0s and 1s. If the ith character in the string is a 0, then there is no snow or ice on the first balcony above the ith lounge chair. If the ith character is a 1, there is unsafe ice or snow on the first balcony above the ith lounge chair.

The third line contains a similar string of N 0s and 1s, indicating the snow and ice conditions on the second row of balconies above each lounge chair.

Output Specification

Please output the number of lounge chairs safe for use (that is, there is no snow or ice on both balconies above the lounge chair).

Constraints and Partial Marks

- For 1 of the 10 available marks, $N = 15$.

- For an additional 9 of the 10 available marks, $N < 1000$.

Sample Input

```
10
0101110101
1101010011
```

Sample Output

```
2
```

Explanation: only the third and seventh lounge chairs are safe. The others all have dangerous snow or ice on at least one of the two balconies above them.

Problem 2: Optimal Skiing

You finally arrive at your dream ski resort after saving up cash for decades. As this vacation was so expensive, you want to spend the maximum amount of time skiing, which means minimizing time on ski lifts.

The mountain is served by many ski lifts. You have studied the map extensively and wrote down your options for taking lifts to the summit of the mountain. Some of these options might involve riding multiple ski lifts. You have also done research on the time it takes to ride each ski lift, in minutes.

Please find the fastest possible time, in minutes, to ascend from the base of the mountain to the summit using one of these ski lift routes.

Input Specification

The first line will consist of an integer N, the number of ski lift routes you can take to get up the mountain.

The next N lines will each describe one of the ski lift routes. The first number in each line will be an integer M, indicating that this route involves taking M lifts. That number will be followed by M integers, representing the travel time for each of the M ski lifts.

Output Specification

Please output one integer, the time required for the shortest ski lift route, in minutes, to get you to the summit. **If this integer is larger than 1000, output the last three digits of this integer.**

Constraints and Partial Marks

- For 1 of the 10 available marks, $N = 7$.
- For 4 of the 10 available marks, $N < 500$, all $M < 16$ and all lifts take at most 30 minutes.
- For the remaining 5 marks, $N < 750$, all $M < 16$ and all lifts take at most 2020 minutes.

Sample Input

```
3
1 10
3 3 5 3
2 1 8
```

Sample Output

```
9
```

Explanation:

The input describes 3 ways to get up the mountain. The first uses one lift that takes 10 minutes. The second uses three lifts, taking 3, 5 and 3 minutes, for a total trip of 11 minutes. The third uses two lifts, taking 1 and 8 minutes, for a total trip of 9 minutes. 9 minutes is the shortest trip.

NOTE: Due to large input sizes, it could take your program a few seconds to read all the input from console and process it.

Problem 3: Farmer Bob

Farmer Bob is trying to move H hay bales (each being 1 meter wide) from one side of his farm to the other.

Farmer Bob has T tractors, each with a certain integer width, in meters. Each tractor can carry as many hay bales as it is wide (i.e. a tractor that is 3 meters wide can carry 3 hay bales at a time), but does not necessarily need to carry a full load each trip.

Farmer Bob's farm is M meters wide. There is a line of trees splitting the farm across the centre. To carry the hay bales across, Farmer Bob's tractor must be able to squeeze through the line of trees without hitting a tree.

Farmer Bob has a map of the tree line. The map is a string of M characters. For each meter across the farm, the map shows a 1 if there are trees in that meter, a 0 if there is only grass in that meter, or an X if there is only gravel in that meter. A tractor can travel through both grass and gravel.

For example, the string 101X01 means that there are trees on the first, third and sixth meters of the tree line, but no trees on the second, fourth or fifth meter.

Farmer Bob's tractors can squeeze through a gap of G meters if the width of the tractor is less than or equal to G. He will use the widest possible tractor that can pass through the line of trees to move the haybales. In the above example, the widest tractor Farmer Bob can use is 2 meters wide (it can squeeze through the gaps between the trees in the third meter and the trees in the sixth meter).

Please determine the least number of trips that Farmer Bob must make to deliver all the H hay bales.

Input Specification

The first line consists of one integer, H, the number of hay bales Farmer Bob must move.

The second line consists of one integer, T, the number of tractors Farmer Bob has.

The third line consists of T integers, the widths of the T tractors, in increasing order.

The fourth line consists of the integer M, the width of Farmer Bob's farm in meters.

The fifth line consists of a string, representing the map of the tree line cutting across the farm.

Output Specification

Please output the minimum number of trips Farmer Bob must make to deliver all the hay bales, assuming he uses the widest tractor he has that can pass through the line of trees. **If this integer is greater than 1000, output the last three digits of this integer.**

Constraints and Partial Marks

- For 1 of the 10 available marks, $H = 120,\ T = 4,\ M = 12$.
- For an additional 4 of the 10 available marks, $H < 10000,\ T < 25,\ M < 500$.
- For the remaining 5 marks, $H < 10^9,\ T < 200,\ M < 1000$.

Sample Input

```
15
3
1 2 4
10
1X1101XX01
```

Sample Output

8

Explanation

The fifth line tells us that the largest gap between trees is 3 meters wide (from meter 7 to meter 9). Out of Farmer Bob's three tractors, the largest one that fits through a 3-meter gap is 2 meters wide (which can carry 2 hay bales each trip). As there are 15 hay bales, Farmer Bob must make 8 trips (carrying 2 hay bales for 7 trips and 1 hay bale for 1 trip).

Problem 4: Bubble Tea

In a group of N friends (including yourself), you travel to the nearest bubble tea store to buy some bubble tea. Everyone picks out their desired bubble tea and lines up to order them. The cashier tells you that there is a special promotion going on: for a purchase of 2 bubble teas, the **cheaper** one is discounted at 25% off, and for a purchase of 3 bubble teas, the **cheapest** one of the three is discounted at 50% off.

You and your friends are already in line, and it is **no longer possible to change the order of this line**. However, you can change how the orders are placed – the 2 or 3 people at the front of the line can group together and place their order together to take advantage of one of the offers (but it is still possible for the person at the front to pay the full price by ordering for themselves only). The way that you order could change the total amount the entire group pays. For example, if you are in a group of 3 and the prices for the 3 people's bubble tea is 100, 100 and 4 (and they are lined up in that order), you can pay in many ways:

Everyone orders themselves: 100 + 100 + 4 = 204 total cost
First two people order together: (100 + 100 – 0.25*100) + 4 = 179 total cost
Last two people order together: 100 + (100 + 4 – 0.25*4) = 203 total cost
Three people order together: (100 + 100 + 4 – 0.50*4) = 202 total cost

In this case, the optimal price for all the bubble tea combined is **179**, which occurs when the first two people place their orders together and the third person orders for themselves.

You and your friends decide to order the bubble teas in a way such that the most money is saved. Please help them **find the minimum amount of money needed for the <u>whole group</u> to buy <u>all</u> the bubble tea optimally** by strategically grouping themselves at the checkout. **For your convenience, all the prices of the bubble tea will be multiples of 4.**

<u>Input Specification</u>

The input will consist of two lines. The first will contain the integer N, the number of friends lined up with bubble tea to buy.

The next N lines will consist of one integer each: the price of each friend's bubble tea, in the order that they lined up to check out.

<u>Output Specification</u>

Output one integer, the price the entire group pays to buy all the bubble tea, assuming they take advantage of the discounts perfectly optimally. **If the optimal price is greater than 1000, output the last 3 digits of the optimal price.**

<u>Constraints and Partial Marks</u>

- For 1 out of the 10 available marks, $N = 5$
- For another 4 out of the 10 available marks, $N < 500$
- For another 5 out of the 10 available marks, $N < 5000$

Sample Input 1

3
100
100
4

Sample Output 1

179

Explanation – Output 1

See the discussion in the problem above.

Sample Input 2

4
40
1000
1000
1000

Sample Output 2

540

Explanation 2

All 4 persons pay for themselves (no discounts): 3040
Persons 1 and 2 go for the 25% discount, persons 3 and 4 go for the 25% discount: 2780
Persons 1 and 2 go for the 25% discount, while persons 3 and 4 pay individually: 3030
Persons 1 and 2 pay individually, while persons 3 and 4 go for the 25% discount: 2790
Persons 2 and 3 go for the 25% discount, while persons 1 and 4 pay individually: 2790
Persons 1-3 go for the 50% discount: 3020
Persons 2-4 go for the 50% discount: 2540

Optimal Price: 2540, of which the last three digits are 540.

NOTE: Due to large input sizes, it could take your program a few seconds to read all the input from console and process it. The optimal solution for this problem allows an execution time of less than 5 seconds. If your program does not produce an answer for an extended period of time, either wait to see if your program eventually produces an answer, or try to optimize your algorithm.

Problem 5: Public Transport

Your brother, who recently broke his leg, needs to take the subway from your house to the pharmacy to get his prescription of pain medication.

This subway system is made of L lines running across the city, numbered 1 through L. Your brother can freely travel from one spot on one line, to any other spot on *the same line*. However, your house and the pharmacy are not on the same subway line, so your brother is going to need to change lines.

Changing lines requires going up or down a floor in a subway station. Unfortunately, due to your brother's broken leg, **he needs the assistance of an escalator to change lines**. Not every line changing station is built with escalators, and some line changing stations only have escalators going in one direction (i.e. for example, it may be possible to go from Line 1 to Line 5, but not from Line 5 to Line 1 using escalators).

As a regular subway user, you have recorded the availability of escalators on the subway system. For example, you may have recorded that there is an escalator allowing a line change from somewhere on Line 1 to somewhere on Line 5 (the actual position on the lines doesn't matter).

Please find the **least number of line changes** your brother must make to get to the pharmacy. (Don't worry about travel times or anything like that, just minimize the number of line changes based on the available escalators to get from the house to the pharmacy). It is guaranteed that your brother is able to make it to the pharmacy.

Note: It is not required (and not possible given the information provided) to visualize the actual layout of subway system to solve the problem. You only need know which line changes are possible to make.

Input Specification

The first line of input will contain 4 space-separated integers. The first integer, L, is the number of lines (numbered 1 through L). The second integer is the subway line that your brother's house is on. The third integer is the subway line that the pharmacy is on. The fourth integer, N, is the number of escalators across the subway system connecting different lines.

The next N lines of input contain two space-separated integers. Each pair of integers represents an escalator, connecting the subway line denoted by the first integer to the subway line denoted by the second integer, and in that direction only.

Output Specification

Output one integer, the least number of line changes your brother needs to do to get to the pharmacy.

Constraints and Partial Marks

- For 1 of 10 available marks, $L = 6, N = 7$
- For the remaining 9 of 10 available marks, $L < 1000$, $N < 1000$ and there are no repeated escalators

Sample Input

```
8 1 6 9
1 2
2 3
1 4
4 7
7 8
3 2
8 6
3 5
6 5
```

Sample Output

```
4
```

Explanation

Your brother starts on Line 1. In order to reach the pharmacy on Line 6, he can take this path:

Line 1 → Line 4 → Line 7 → Line 8 → Line 6.

This path takes four transfers. It is easy to see that this path requires the least number of transfers.

NOTE: Due to large input sizes, it could take your program a few seconds to read all the input from console and process it. The optimal solution for this problem allows an execution time of less than 5 seconds. If your program does not produce an answer for an extended period of time, either wait to see if your program eventually produces an answer, or try to optimize your algorithm.