## Problem 4 – Dancing Bits

Gergana loves dancing and she also likes bits (she doesn't know what bits really are, but she knows that she likes them). Few days ago she accidently invented a new term - “**dancing bits**”.

If you ask her what “dancing bits” mean she will tell you that it’s a sequence of identical bits (so the bits can dance together – zeros can only dance with other zeros, the same applies for ones).

You are given **N** positive integer numbers that are converted to binary numeral system and are concatenated together in one big sequence of bits.

For example: if we have 4 numbers: **5** (101 in binary numeral system), **6** (110 in binary numeral system), **14** (1110 in binary numeral system) and 143 (1000111 in binary numeral system) their concatenation will be **101110111010001111.**

You are also given a positive integer **K** - the number of identical bits (zeroes or ones that can dance together).

Write a program that finds the number of all “dancing bits” (the sequences of equal bits) with a length of exactly **K bits**. Your program should search in the concatenation of the given **N** numbers.

For example, if we have **4** numbers (**5**, **6**, **14** and **143**, the concatenation of their binary representation is 10**111**0**111**01**000**1111) and we are searching for the total number of all sequences of equal bits with an exact length of **3** bits, the answer will be **3** (the sequences are bolded in the concatenation above).

In this example we have two sequences of “dancing bits” - "111" consisting of only ones and one sequence of “dancing bits” - "000" consisting of only zeros. Note that the sequence "1111" is not a sequence of exact 3 identical bits.

### Input

The input data should be read from the console.

At the first input line there will be one positive integer – the number **K**.

At the second input line there will be another positive integer – the number **N**.

At each of the next **N** lines there will be one positive integer – the N numbers that represent the input sequence of bits.

The input data will always be valid and in the format described. There is no need to check it explicitly.

### Output

The output data should be printed on the console.

The only output line must contain the answer – the number of “dancing bits” sequences found.

### Constraints

* The number **K** will be positive integer number between 1 and 25 600, inclusive.
* The number **N** will be positive integer number between 1 and 800, inclusive.
* Each of the **N** numbers will be positive integer numbers between 1 and 2 147 483 647, inclusive.
* Allowed working time for your program: 0.20 seconds.
* Allowed memory: 16 MB.

### Examples

|  |  |
| --- | --- |
| **Input example** | **Output example** |
| 3  4  5  6  14  143 | 3 |
| 1  4  2  10  42  170 | 20 |