

## Problem I. Submatrix Search

Input file:            submatrix.in  
Output file:           submatrix.out  
Time limit:           10 seconds  
Memory limit:         256 megabytes

You are given a  $10^6$  by  $10^6$  square matrix of zeroes and ones. The number in row  $i$  ( $1 \leq i \leq 10^6$ ) and column  $j$  ( $1 \leq j \leq 10^6$ ) has its value defined by the following Java code (it's called *Jenkins one-at-a-time hash*):

```
long where = i * 1000000L + j;
int hash = 0;
for (int k = 0; k < 5; ++k) {
    hash += (int) ((where >>> (8 * k)) & 255);
    hash += (hash << 10);
    hash ^= (hash >>> 6);
}
hash += (hash << 3);
hash ^= (hash >>> 11);
hash += (hash << 15);
return (hash >>> 27) & 1;
```

The  $\ggg$  operator is “unsigned shift right”, “long” type is 64-bit integer, and “int” is 32-bit integer. Note that this hash function was selected somewhat arbitrarily, and this problem would be solvable for any sufficiently random  $f(i, j)$  defining the values.

Then we randomly and uniformly pick two numbers  $a$  and  $b$  ( $1 \leq a, b \leq 10^6 - 10^3 + 1$ ), and write down the 1000 by 1000 submatrix in rows between  $a$  and  $a + 999$ , inclusive, and in columns between  $b$  and  $b + 999$ , inclusive.

Can you find  $a$  and  $b$  given the submatrix? You don't need to find the original  $a$  and  $b$ , any pair that would yield the given submatrix will do.

### Input

The input file contains 1000 lines. Each line corresponds to one row of the submatrix and contains 1000 characters.

### Output

Output two space-separated numbers  $a$  and  $b$ .

## Examples

| submatrix.in   | submatrix.out |
|--|---------------|
| 0000100111<br>1110101010<br>1010010001<br>0001101110<br>1001100010<br>0111011100<br>1001010001<br>0110110110<br>1010010110<br>0001011011 | 165731 269905 |

## Note

Note that the first test is bigger than the example above, since it has 1000 lines and 1000 characters in each line. You can download the first test at <http://forest.acm.petrsvu.ru/tests/submatrix.in>. This problem has 20 testcases, each generated randomly according to the description from the problem statement.