ALGORITHM DESIGN TERM PROJECT

1506. Columns of Numbers

Parichart Meesin	6010017
Kamonrut Vorrawongprasert	6013633

Introduction

Every New Russian has to look through long columns of numbers for analyzing market trends and planning his investments. Psychologists assure that the longer is a column of numbers, the more difficult it is to perceive it. Therefore, it is better to print numbers not in one long column, but in several columns so that their height would be minimal. Transform a given sequence of numbers to a format that is psychologically more convenient for perception.

Time limit: 1.0 second

Memory limit: 64 MB

Difficulty: 156

Problem

Input

The first line contains two integers: N (1 ≤ N ≤ 100), which shows how many numbers must be analyzed, and K (1 ≤ K ≤ N), which is the desired number of columns. The second line contains N integer numbers in the range from 0 to 999.

Output

Output the N numbers given in the input in K columns in such a way that the number of lines is minimal and the columns have the same height with the possible exception of the last column, which may be shorter. The width of each column must be 4 symbols; the numbers must be aligned to the right edge and padded with spaces to the required width. The numbers must be given in the same order as in the input, but in columns: the first column from the top to the bottom, then the second column from the top to the bottom, and so on. All nonempty lines must end with a line break; there must be no end spaces in the lines. It is guaranteed that solution is always exist.

Input

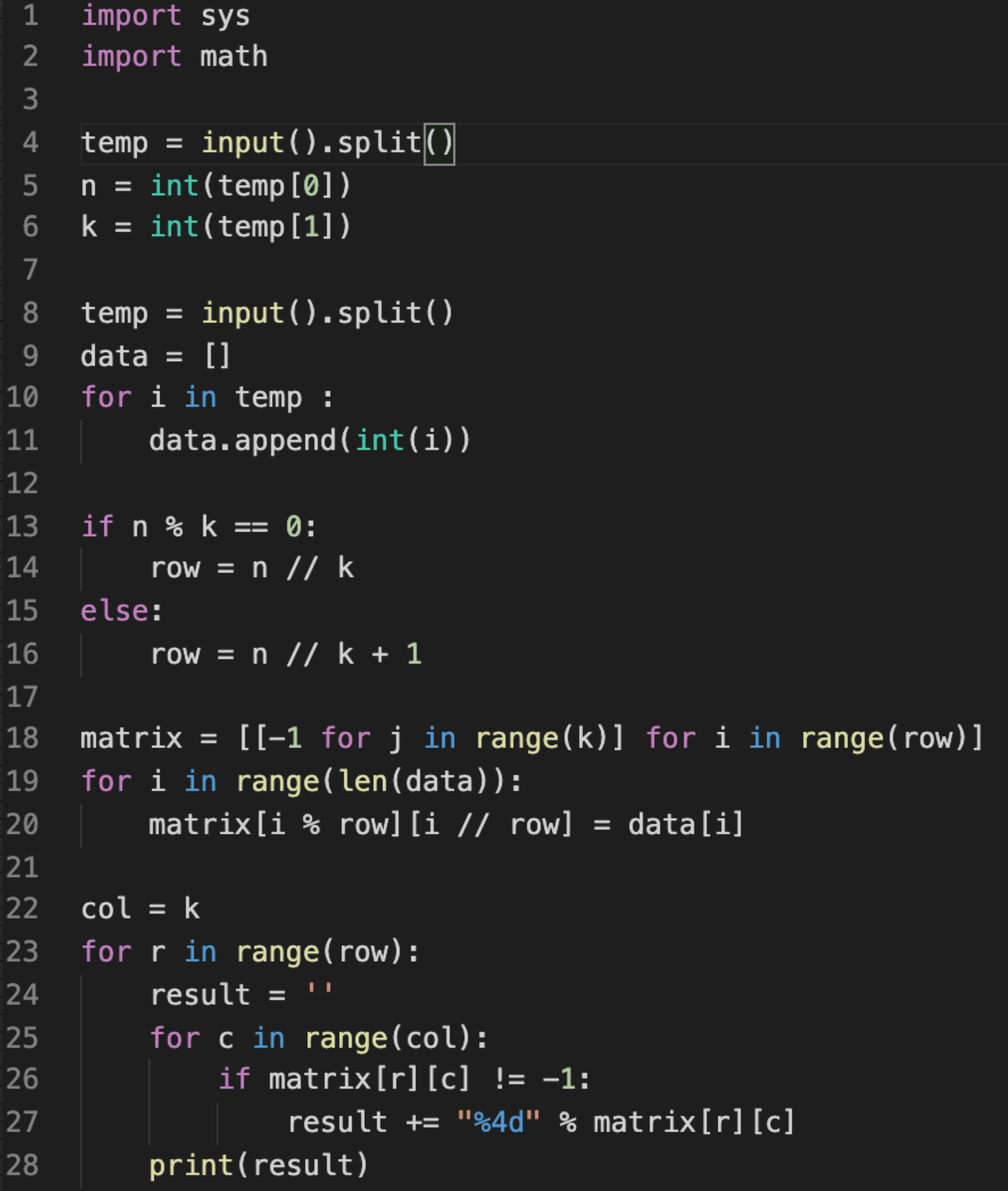
```
7 3
1 2 30 40 50 600 700
```

Output

```
   1  40 700
   2  50
  30 600
```

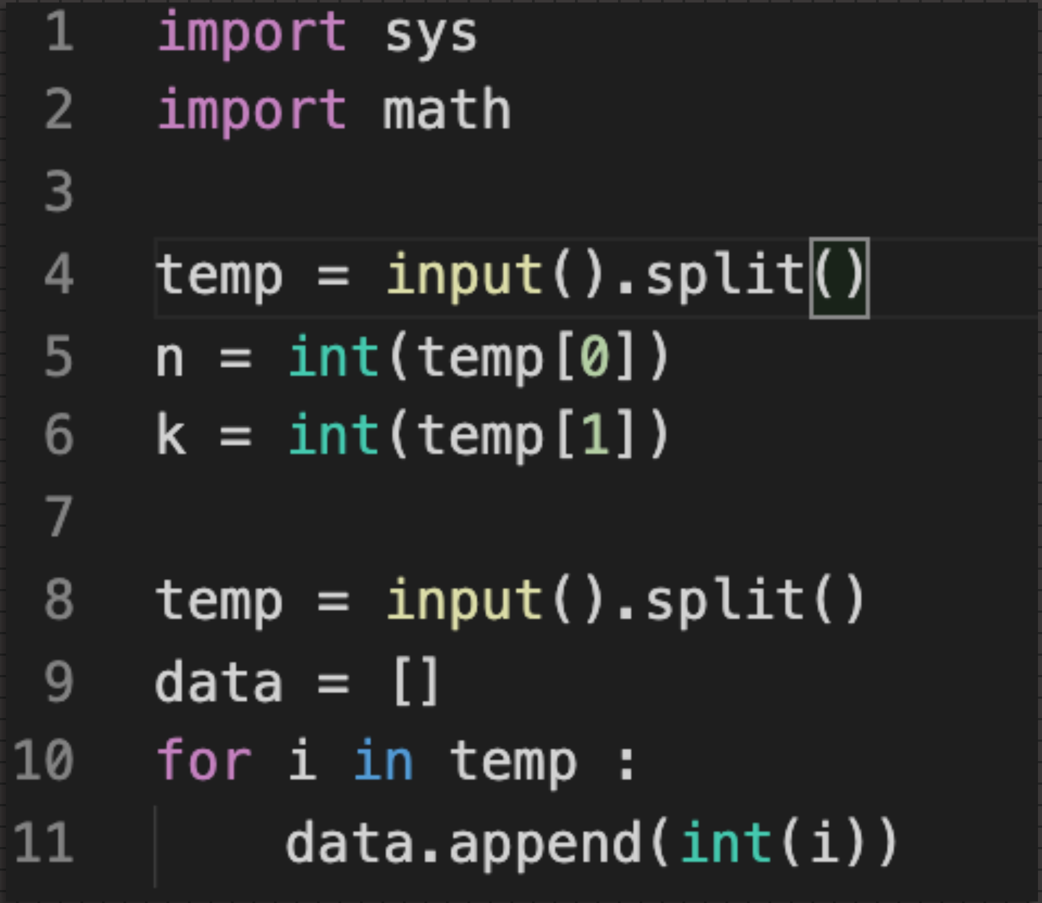
Solution

We use the matrix to allocate the set of number into column and row, also use mod and floor division to find the rows and column that the number should be out in.



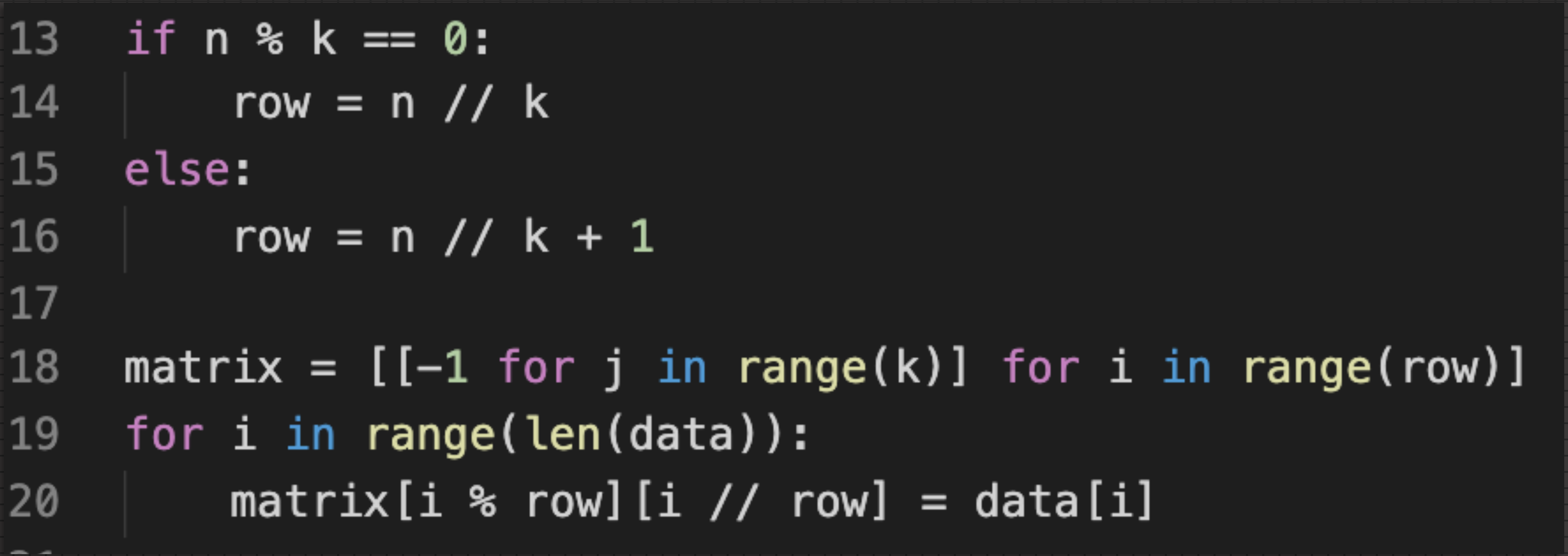
```python
import sys
import math

temp = input().split()
n = int(temp[0])
k = int(temp[1])

temp = input().split()
data = []
for i in temp :
    data.append(int(i))

if n % k == 0:
    row = n // k
else:
    row = n // k + 1

matrix = [[-1 for j in range(k)] for i in range(row)]
for i in range(len(data)):
    matrix[i % row][i // row] = data[i]

col = k
for r in range(row):
    result = ''
    for c in range(col):
        if matrix[r][c] != -1:
            result += "%4d" % matrix[r][c]
    print(result)
```

Code



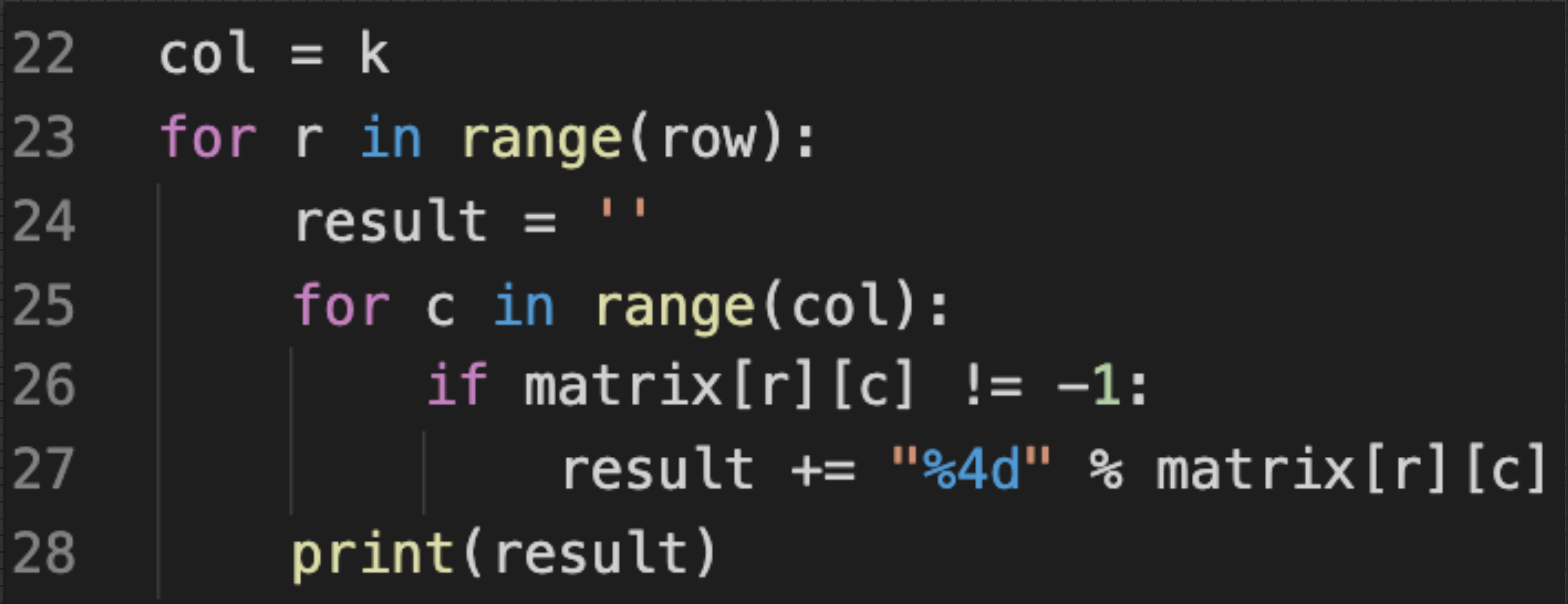
```python
import sys
import math

temp = input().split()
n = int(temp[0])
k = int(temp[1])

temp = input().split()
data = []
for i in temp :
    data.append(int(i))
```

Code



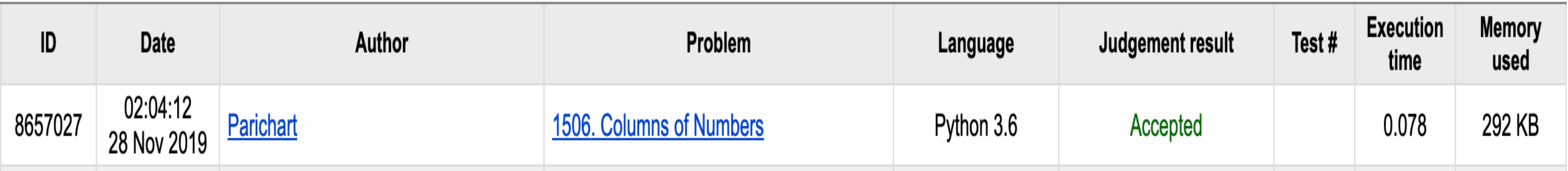
```python
if n % k == 0:
    row = n // k
else:
    row = n // k + 1

matrix = [[-1 for j in range(k)] for i in range(row)]
for i in range(len(data)):
    matrix[i % row][i // row] = data[i]
```

Code



```python
col = k
for r in range(row):
    result = ''
    for c in range(col):
        if matrix[r][c] != -1:
            result += "%4d" % matrix[r][c]
    print(result)
```

Result



ID	Date	Author	Problem	Language	Judgement result	Test #	Execution time	Memory used
8657027	02:04:12 28 Nov 2019	Parichart	1506. Columns of Numbers	Python 3.6	Accepted		0.078	292 KB

Reference

https://github.com/fanofxiaofeng/timus/blob/master/1506/main.py

Thank you