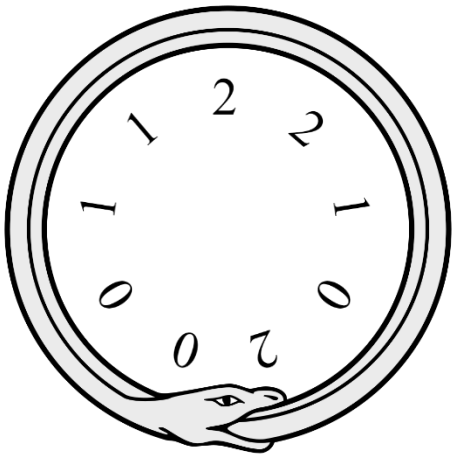
# Ouroboros

In Egyptian mythology, Ouroboros is a serpent that is depicted consuming its own tail. It is often used as a motif to symbolise fertility and the renewal of life. However, the Guild of Sorcerers are far more interested in the symbol's alchemical properties.

An Ouroboros Ring is a circular sequence of numbers that contains all the permutations of digits 0 to *n-1* of length *m*. For example, the Ouroboros Ring below contains all permutations of the digits 0, 1 and 2 of length 2.




Starting from the top we can read clockwise to get all permutations:

22, 21, 10, 02, 20, 00, 01, 11, 12.

It is also a minimum ring – it has the shortest possible length.

We can represent this ring linearly by starting at any point and reading clockwise until we get back where we started: 221020011.

Write a program to construct minimum Ouroboros Rings for the Guild. You will be given 2 integers, $n$ ($2 \leq n \leq 9$) and $m$ ($2 \leq m \leq 15$), separated by spaces, which denote the number of digits in the ring and the length of the permutations respectively. The length of the minimum ring will never exceed 10,000. You should output a single sequence of digits, representing the desired Ouroboros ring.

*Note that there are multiple valid solutions for each input.*

Sample Input 1:

```
3 2
```

Sample Output 1:

```
221020011
```

Sample Input 2:

```
2 4
```

Sample Output 2:

```
0000111101100101
```

Sample Input 3:

```
5 2
```

Sample Output 3:

```
1223344324214131
040302001
```