

Problem B. Balanced Illumination

Time limit: 1 second
Memory limit: 512 megabytes

Saint Bitsburg government is preparing a technical requirement for New Year city decoration.

The governor thinks that there should be a garland of n lights on the main square. The lights will turn on and off and entertain the residents of Saint Bitsburg.

The chief designer decided that the garland would change its appearance each second. Every light in the garland can be in two states: on and off. Each second exactly one light will change its state from on to off, or from off to on. Also the chief designer wants all combination of lights in the garland to repeat with a period 2^n seconds. During the period of 2^n seconds all 2^n possible lights combinations in the garland have to be presented.

The city's chief engineer, however, noted that frequently turning lights on and off would cause their malfunction. To minimize the chance of the lights malfunction it is required for every light to be turned on and off approximately the same number of times.

So, the final technical requirement for you — Chief Programmer of the Government Department of Information Technology — is here.

- You need to make a plan of 2^n combinations of lights $a_0, a_1, \dots, a_{2^n-1}$, where a_k is a line of n zeros and ones, $a_k[i] = 1$ means, that the light i in the combination a_k is on, $a_k[i] = 0$ means, that the light i in the combination a_k is off.
- All combinations in the plan have to be distinct.
- This plan will be launched in a cycle, each second the next combination is presented on the garland, in the t -th second the combination $a_{t \bmod 2^n}$ is presented.
- Adjacent combinations have to differ in exactly one light's state. Combination a_{2^n-1} and a_0 also have to differ in exactly one light's state.
- Let us denote by c_i the number of state changes of the light i during a complete cycle, including the final change from a_{2^n-1} to a_0 . Then for any $i \neq j$ values c_i and c_j have to differ by no more than 2.

Get to work!

Input

Input contains one integer n ($1 \leq n \leq 17$).

Output

Output 2^n lines of n characters — sequence of combinations in the plan. It is guaranteed that a plan satisfying all requirements exists.

Examples

standard input	standard output
3	000 010 011 111 110 100 101 001
4	0000 0010 1010 1011 0011 0111 0110 0100 0101 0001 1001 1101 1111 1110 1100 1000

Note

In the first sample test $c_1 = c_2 = 2$, $c_3 = 4$.

In the second sample test $c_1 = c_2 = c_3 = c_4 = 4$.