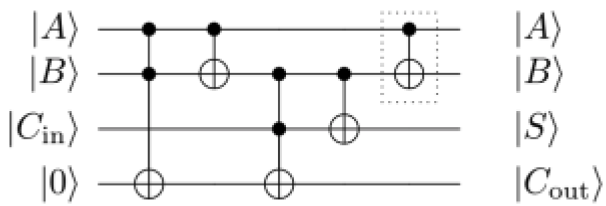
Problem C. Matrix Logic Circuits

time_limit 500 ms
mem_limit 1048576 kB
os Windows

In quantum computing, logic gates operate a bit differently. Quantum logic gates are reversible, and the number of input qubits is equal to the number of output qubits. Additionally, they can be represented by 2^N by 2^N matrices, where N is the number of qubits.

A quantum circuit is a model for quantum computation where the computation is performed through a sequence of quantum logic gates and measurement devices. A sequence of logic gates can be represented by a matrix resulting from the multiplication of the matrices of the logic gates in the order of application, which is the reverse order of how they are graphically represented. For example, the circuit for adding two bits in its quantum form is:



In this circuit, we have two variations of a logic gate that we will call $\text{CNOT}(q_c, q_t)$ and $\text{CCNOT}(q_{c_1}, q_{c_2}, q_t)$. In the diagram, the qubit q_t is marked with a $\oplus$. The logic gate $\text{CNOT}(q_c, q_t)$ can be seen as being equal to $\text{CCNOT}(q_c, q_c, q_t)$, that is, the application of the logic gate CCNOT with $q_c = q_{c_1} = q_{c_2}$.

The logic gate $\text{CCNOT}(q_{c_1}, q_{c_2}, q_t)$ behaves by inverting the output qubit q_t if both control qubits q_{c_1} and q_{c_2} are set. Mathematically, $q_t' = q_t \oplus (q_{c_1} \wedge q_{c_2})$. In its matrix form:

$$\text{CCNOT}(q_{c_1}, q_{c_2}, q_t)_{ij} = \begin{cases} 1 & \text{if } i \text{ has bits } c_1 \text{ and } c_2 \text{ set and } i \oplus 2^t = j \\ 0 & \text{if } i \text{ has bits } c_1 \text{ and } c_2 \text{ set and } i \oplus 2^t \neq j \\ 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}$$

where i is the row and j is the column with $0 \leq i, j < 2^N$, and i contains bit k $(0 \leq k)$ if $\left\lfloor \frac{x}{2^k} \right\rfloor \mod 2 = 1$. The operation $\oplus$ is the bitwise exclusive or operation, commonly represented as ^ in programming languages.

Thus, the matrix of the quantum circuit for adding two bits is given by

$$\mathrm{CNOT}(q_0, q_1)\,\mathrm{CNOT}(q_1, q_2)\,\mathrm{CCNOT}(q_1, q_2, q_3)\,\mathrm{CNOT}(q_0, q_1)\,\mathrm{CCNOT}(q_0, q_1, q_3),$$

where the qubits q_0, q_1, q_2, q_3 are used with inputs $|A\rangle, |B\rangle, |C_{in}\rangle, |0\rangle$ respectively and result in $|A\rangle, |B\rangle, |S\rangle, |C_{out}\rangle$ respectively.

Your task is given the description of a circuit with logic gates CNOT and CCNOT in the order of application, to print the resulting matrix.

Input

The first line of the input contains the integers N ($2 \leq N \leq 8$), the number of qubits in the circuit, and M ($1 \leq M \leq 10^5$), the number of logic gates in the circuit.

This is followed by M lines, each with the description of a logic gate. The first integer T ($1 \leq T \leq 2$) defines the type of the logic gate. If $T = 1$, the description is for the logic gate $\mathrm{CNOT}(q_C, q_T)$ and is followed by distinct integers C and T ($0 \leq C, T < N$). If $T = 2$, the description is for the logic gate $\mathrm{CCNOT}(q_{C1}, q_{C2}, q_T)$ and is followed by distinct integers $C1$, $C2$, and T ($0 \leq C1, C2, T < N$). Note that the logic gates are given in the order of application.

Output

Print 2^N lines, each containing exactly 2^N characters '0' or '1', corresponding to the complete matrix of the quantum circuit.

Examples

Input	Output
2 1 1 0 1	1000 0001 0010 0100

Input	Output
4 5 1 0 1 1 1 2 2 1 2 3 1 0 1 2 0 1 3	1000000000000000 0000000000000100 0000000000000010 0000000000010000 0000100000000000 0100000000000000 0010000000000000 0000000000000001 0000000010000000 0000010000000000 0000001000000000 0001000000000000 0000000000001000 0000000001000000 0000000000100000 0000000100000000

Input	Output
3 1 2 0 1 2	10000000 01000000 00100000 00000001 00001000 00000100 00000010 00010000

Input	Output
3 1 1 0 1	10000000 00010000 00100000 01000000 00001000 00000001 00000010 00000100