

Cycle

SEASON 10 – SIXTH ROUND



Given is an undirected graph with N vertices and M edges. Each of the edges has a value w_i . Let us consider all cycles, which start from a vertex, cross a sequence of edges (each edge should be crossed no more than once) and their last vertex coincides with the first one. Let denote the values of the edges included in one such cycle C with e_1, e_2, \dots, e_k . Then the value of the function $F(C) = \max(e_1, e_2, \dots, e_k) + \min(e_1, e_2, \dots, e_k)$.

Write a program which finds the maximal value of F for any of the cycles in the graph.

Input

The first line of the input file `cycle.in` contains the natural numbers N and M . Each of the following M lines contains three integers, separated by a single space – u_i, v_i and w_i – the two vertices connected by the edge and its value. It is possible that between some pairs of vertices there are multiple edges. There are not edges which connect a vertex to itself.

Output

On the only line of the output file `cycle.out` print a single number, equal to the maximal value of the function F . If the graph does not contain cycles, print 0.

Constraints

$$1 \leq N, M \leq 10^5$$

$$0 \leq w_i \leq 10^9$$

$$1 \leq u_i, v_i \leq N$$

Example

Input	Output
7 7 1 2 5 1 3 3 2 3 4 2 4 1 3 4 8 5 6 3 5 6 7	10