**Maximum Flow**

**The Problem**

Given a network which has a single source vertex with no in edges and a single destination (sink) vertex with no out edges, compute the maximum flow capacity for the network.

**The Input**

The first line contains the number of test cases *T* (*T* <= 250).

The first line of each test case contains 4 integers – *N*, *E*, *S* and *D*. *N* is the number of vertices in the network, 2 ≤ *N* ≤ 1000. *E* is the number of directed edges in the network, 1 ≤ *E* ≤ 10,000. *S* is the source vertex of the network, 0 ≤ *S* < *N*. *D* is the sink vertex of the network , 0 ≤ *D* < *N*. There is a further constraint that *S* ≠ *D*.

The next *E* lines contain three integers – *Vi*, *Vj*, *C* – where *Vi* is the source vertex of the directed edge, *Vj* is the destination vertex of the directed edge and *C* is the capacity of the directed edge. 1 ≤ *C* ≤ 10,000.

**The Output**

For each test case, output the message “Test x: Maximum flow = y”, where x is the test case number and y is the maximum flow capacity for the given network. Test case numbers start at 1.

**Sample Input**

1

7 8 0 6

0 1 3

0 2 1

1 3 3

2 3 5

2 4 4

3 6 2

4 5 2

5 6 3

**Sample Output**

Test 1: Maximum flow = 3