Flow contest editorial

Ivan Smirnov
ifsmirnov@yandex.ru
t.me/ifsmirnov

September 23, 2019
Discover Singapore 2019

A. Just a flow

Well, just find a maximum flow by definition.

B. The Cut

After finding the maximum flow run a DFS from the source in the residual network. Now the cut between visited vertices and all other graph is the minimum cut by max-flow-min-cut theorem (recall its constructive proof).

C. Flow decomposition

Find the maximum flow. Now run successive DFSes from source to sink, following only edges with nonzero flow. When the sink is reached, print the found path and subtract the value of minimum flow on this path from all its edges' flows. After several such operations the remaining flow will consist only of several cycles. They may be discarded as cycles do not affect the flow value.

D. Snails

Set the capacity of each edge to 1 and find the flow of size 2 from s to t (that is, make two runs of the FF algorithm). Indeed, each path for snails is the flow of value 2 because it is effectively two edge-disjoint paths.

E. Matan

A variation of the closure problem.

Make artificial source and sink. There will be 3 types of edges:

1. edge of infinite capacity from each vertex to its preliminaries

2. edge of capacity x from source to vertex with positive usefulness x
3. edge of capacity x from vertex with negative usefulness $-x$ to a sink

Now the answer is (the sum of all positive usefulnesses) minus (the maximum flow in this graph). For the proof, try understanding what is the meaning of any cut in this graph.

F. Perspective

Of course, our team should win all possible matches, and all other teams should lose all the matches with other division. Now we have to check whether it is possible to distribute victories in matches within our division such that no team scored more than d.

Make a vertex for each team with edge of capacity d to a sink. For each pair of teams, make a vertex for their "confrontation" with edge of capacity "number of matches pending" from source to it. Next, add edges of infinite capacity from each "confrontation" to both involved teams.

Now the answer is "YES" if and only if the maximum flow in this graph saturates all edges from the source. Indeed, the flow splits the internal matches between teams, and the "complete" flow exists only if it is possible to satisfy all constraints.

G. Full operation

Binary search by the answer. Let's check if we can have the outdegree d or less.

Make a graph on $n+m+2$ vertices: n for vertices of the initial graph, m for its edges, a source, and a sink. Now add a unit capacity edge from each edge to its endpoints and from the source to each edge, and an edge of capacity d from each vertex to sink. Now the answer to the subproblem is positive if the maximum flow in this network is m.

Indeed, if the flow has value m then it passes through each initial edge. For each edge, it contributes to either of its endpoints. And none of the endpoints was contributed to more than d times due to flow restrictions.

H. Looking for brides

This was a problem on a min-cost flow, so don't worry if you didn't solve it.

If we build a network where each edge of the initial graph has unit capacity and corresponding cost, then the answer to the problem will be the k-th part of the min-cost flow of size k. Indeed, the flow of size k in such graph corresponds for k paths which do not share an edge. And min-cost ensures that the average time (or the total time, which is essentially the same) is minimized.