2ND PRACTICAL HOMEWORK

Optimization methods 2017/18

Independence in the school acquaintance graph

You have a large graph on input which describes acquaintances at a school. Imagine a large school with several classes, where vertices are pupils and edges correspond to who knows whom – this relation being symmetric in our case.

The school acquaiantance graph on input is not uniformly random. It always contains several large classes – groups of pupils where everyone knows each other. As in real life, there are pupils who know pupils from other classes, but those edges are relatively scarce in the school acquaintance graph.

Your task is to generate a linear/integer program that finds a *maximum independent set* – the largest group of pupils where nobody knows each other.

The important condition is that your solution should be relatively fast. For the inputs given in our archive, your IP/LP should run in the matter of *seconds*, not minutes (on a fairly recent computer).

Input format

The file with the undirected graph has the following format: The first line contains "`n m`"where n is the number of vertices and m is the number of edges. Vertices are indices by $0, \ldots, n-1$. Next m lines contains "`i j`"giving end vertices of all edges.

Example (graph $K_{1,2}$):

```
3 2
0 1
0 2
```

Output format

The output of a program must contain the following section which starts by the following line.
`#OUTPUT:`
Next line must contain the optimal number t of vertices in a maximum independence set followed by t lines listing all vertices of one maximum independence set. The result section must be terminated by the following line.
`#OUTPUT END`

Output for the example input.

```
#OUTPUT:
2
1
2
#OUTPUT END
```

Evaluation

Students are expected to submit source codes of their program and a document in PDF describing their approach (usually linear programming model used to solve given task). Submitted programs must be able to find an optimal solution and print it described format for every possible input. Buggy programs failing this condition are evaluated by 0 points. The number of points from this homework mainly depends on size of testing input a program solves on a testing computer in a given time. Testing inputs will be similar as example input available on lecture's website.