Discrete Optimization - Assignment 2

In this assignment you will be asked questions related to the (weighted) set covering problem (SCP) and the use of approximation algorithms and heuristics for solving it.

The assignment is due 23:59 on 27.10.2017 and should be completed in groups of 2 or 3. Please upload your written report and source code to Absalon. You are permitted to discuss the assignment with students from other groups but should not copy from each other.

1 Theoretical part

Consider the randomized rounding algorithm for Set Cover (Section 14.2 in Vazirani). The algorithm picks a collection $\mathcal{C}$ of sets. Show that with some constant probability, $\mathcal{C}$ covers at least half the elements at a cost of at most $O(OPT)$.

2 Implementation part

You should attempt to solve the 4 instances which have been uploaded to Absalon with the methods listed below. For each method and problem instance, list the best solution value obtained and computation time in a table.

2.1 CPLEX

A CPLEX model has been uploaded to Absalon and you should either use this or make your own model to attempt to solve the instances. The largest problem instance should not be solvable to optimality using this method.

The CPLEX model is represented using the OPL language in a .mod file. See documentation here

- `https://www.ibm.com/support/knowledgecenter/SSSA5P\_12.6.3/ilog.odms.studio.help/pdf/opl\_languser.pdf`
- `https://www.ibm.com/support/knowledgecenter/SSSA5P\_12.6.2/ilog.odms.studio.help/pdf/opl\_langref.pdf`

The problem instances are in .dat files. You can use the

```
<CPLEXDIR>/opl/bin/<arch>/oplrun
```

application to find a solution given a .mod and a .dat file. Note that this application only appears in the linux and windows versions of the CPLEX Studio.

2.2 Rounding

Solve the problem instances using simple rounding (Algorithm 14.1 in Vazirani) *and* using randomised rounding (Section 14.2 in Vazirani). Relate the experimental results of randomised rounding to the theoretical question given above. You can use any programming language you want but we recommend using the CPLEX library to solve the LP (for example by including it in a java-project as in the first assignment).

2.3 Third method

Solve the problem instances using the primal-dual method (Chapter 15 in Vazirani) *or* a metaheuristic of your choosing.