Course 02249, DTU Compute
Carsten Witt

Computationally Hard Problems – Fall 2016
Assignment 5

Date: 27.09.2016, **Due date**: 03.10.2016, 21:00

For all problems on this sheet, you may assume that they are in $\mathcal{NP}$.

The following exercises are **not** mandatory:

Exercise 5.1: Prove that the transformation T in the proof of Theorem 4.19 from the lecture notes can be performed in time polynomial in the input size (which is the total length of all clauses plus the number of variables).

End of Exercise 1

Exercise 5.2: We consider the following problem.

Problem: [4-SAT]
Input: A set of clauses $C = \{c_1, \dots, c_k\}$ over n boolean variables $x_1, \dots, x_n$, where every clause contains exactly four literals.
Output: YES if there is a truth assignment to the variables $a\colon \{x_1, \dots, x_n\} \to \{0, 1\}$ such that every clause c_j is satisfied. The answer is NO otherwise.

Prove that 4-SAT is $\mathcal{NP}$-complete.
Note: A clause is a set of literals, i. e., it is not allowed to have the same literal multiple times in a clause.

End of Exercise 2

Exercise 5.3: Recall the following problem, which was already defined on Exercise Sheet 2.

Problem: [REFUTATION] Given is a disjunctive form consisting of k monomials $m_1, \dots, m_k$ over n boolean variables $x_1, \dots, x_n$. The task is to decide if there is a truth assignment to the variables $a\colon \{x_1, \dots, x_n\} \to \{0, 1\}$ such that the truth value of the disjunctive form is *false*.

Prove that REFUTATION is $\mathcal{NP}$-complete.
Hint: De Morgan's laws from propositional logic can be useful.

End of Exercise 3

Continued on next page.

Exercise 5.4: Consider the following problem:

Problem: [ZeroSum]
Input: A sequence $a_1, \ldots, a_n$ of integers.
Output: YES if there is a subset $S \subseteq \{1, \ldots, n\}$ which is non-empty and where the integers sum to 0, i. e.,

$$S \neq \emptyset \text{ and } \sum_{i \in S} a_i = 0,$$

and NO otherwise.

Prove that the problem ZeroSum is $\mathcal{NP}$-complete. To do this, you may use that PotatoSoup is $\mathcal{NP}$-complete.

End of Exercise 4

The following exercise is **mandatory**:

Exercise 5.5: Consider the following problem.

Problem: [Sat-Two-Thirds]
Input: A set of clauses $C = \{c_1, \ldots, c_k\}$ over n boolean variables $x_1, \ldots, x_n$.
Output: YES if there is a truth assignment to the variables such that at least $(2/3)k$ many clauses are satisfied. NO otherwise.

Show that this problem is $\mathcal{NP}$-complete. You may use any problem stated as $\mathcal{NP}$-complete in the lecture notes for this course. You may also assume that Sat-Two-Thirds is in $\mathcal{NP}$.
Especially show:

(A) Find a suitable problem P_c which is known to be $\mathcal{NP}$-complete.

(B) Prove $P_c \leq_p$ Sat-Two-Thirds, especially:

- (B.1) Describe a transformation T which transforms every instance $\boldsymbol{X}$ of P_c into an instance $T(\boldsymbol{X})$ of Sat-Two-Thirds and which runs polynomial in the size $|\boldsymbol{X}|$ of $\boldsymbol{X}$.
- (B.2) Show that if the answer to $\boldsymbol{X}$ is YES then so is the answer to $T(\boldsymbol{X})$.
- (B.3) Show that if the answer to $T(\boldsymbol{X})$ is YES then so is the answer to $\boldsymbol{X}$.

Hint: If you decide to reduce from 3-Sat, you may want to introduce some extra variables and clauses of length 1.

End of Exercise 5