Formale Methoden der Informatik
Block 1: Computability and Complexity

Exercises 1-10

SS 2012

Exercise 1 *Consider the problem* ***PROCEDURE NEG-ASSIGNMENT****, which is defined as follows:*

> ***PROCEDURE NEG-ASSIGNMENT***
>
> *INSTANCE: A triple* (Π, I, k)*, where (i)* Π *is a program that takes one string as input and outputs true or false, (ii)* I *is a string, and (iii)* k *is an integer variable used in program* Π*.*
> *QUESTION: Does variable* k *ever get assigned a negative value when the program* Π *is executed with input* I*?*

Prove that ***NEG-ASSIGNMENT*** *is undecidable. Prove the undecidability by providing a reduction from the* ***HALTING*** *problem to* ***NEG-ASSIGNMENT****, and arguing that your reduction is correct.*

Exercise 2 *Prove that the problem* ***NEG-ASSIGNMENT*** *from Exercise 1 is semi-decidable. To this end, provide a semi-decision procedure and justify your solution. Additionally, show that the co-problem of* ***NEG-ASSIGNMENT*** *is not semi-decidable.*

Exercise 3 *Give a formal proof that* ***SUBSET SUM*** *is in* NP*, i.e. define a certificate relation and discuss that it is polynomially balanced and polynomial-time decidable.*

In the ***SUBSET SUM*** *problem we are given a finite set of integer numbers* $S = \{a_1, a_2, \ldots, a_n\}$ *and an integer number* t*. We ask whether there is a subset* $S' \subseteq S$ *whose elements sum is equal to* t*?*

Exercise 4 *Formally prove that* ***PARTITION*** *is* NP*-complete. For this you may use the fact that* ***SUBSET SUM*** *is* NP*-complete.*

In the ***PARTITION*** *problem we are given a finite set of integers* $S = \{a_1, a_2, \ldots, a_n\}$*. We ask whether the set* S *can be partitioned into two sets* S_1, S_2 *such that the sum of the numbers in* $S1$ *equals the sum of the numbers in* S_2*?*

Exercise 5 *Formally prove that* ***FREQUENCY ASSIGNMENT*** *is* NP*-complete. For this you may use the fact that a similar problem used in lectures is* NP*-complete.*

In the ***FREQUENCY ASSIGNMENT*** *problem we are given a set of transmitters* $T = \{t_1, t_2, \ldots, t_n\}$*,* k *frequencies, and the list of pairs of transmitters that interfer and therefore cannot use the same frequency. We ask whether there is an assignment of each transmitter to one of* k *frequencies such that there is no interference between the transmitters.*

Exercise 6 *Fomally prove that logical entailment is* co $-$ NP*-complete. The formal definition of entailment* ($\models$) *is this:* $\alpha \models \beta$ *if and only if, in every truth assignment in which* α *is true,* β *is also true.*

Exercise 7 *It is well known that the* ***k-COLORABILITY*** *problem is* NP*-complete for every* $k \geq 3$*. Recall that the instance of* ***k-COLORABILITY*** *is an undirected graph* $G = (V, E)$*. Suppose that we restrict this instance of* ***k-COLORABILITY*** *to trees. Can the restricted problem be solved with an algorithm that runs in polynomial time? If yes, provide such an algorithm.*

Exercise 8 *Provide a reduction of* ***N-Queens*** *problem to* ***SAT****. Give a proof sketch of the correctness of your reduction. Does this implies that the* ***N-Queens*** *is an* NP*-complete problem? Argue your answer.*

In the ***N-Queens*** *problem we are given* n *queens and an* $n \times n$ *chessboard. We ask whether we can place these* n *queens on the chessboard such that no two queens attack each other. Two queens attack each other if they are placed in the same row, or in the same column, or in the same diagonal.*

Exercise 9 *Consider the following problem:*

N-SORTED-ELEMENTS
INSTANCE: A non-empty list $L = (e_1, \ldots, e_n)$ *of non-negative integers.* *QUESTION: Does the list* L *contain a sub-list of* k *consecutive sorted numbers in ascending order (from left to right)?*

Argue that ***N-SORTED-ELEMENTS*** *can be solved using only logarithmic space.*

Exercise 10 *Design a Turing machine that increments by one a value represented by a string of 0s and 1s.*