Exercise session 10

1. Let's consider a special variation of Turing machines, with all states classified as "bell-states" or "whistle-states". In each state the machine either rings the bell or blows the whistle, depending on the type of state. Otherwise the machines are ordinary Turing machines. Show that it is undecidable whether the given machine M will ever blow the whistle with given input w!

2. Are the following problems solvable, partially solvable or totally unsolvable?

   a) Does the given Turing machine halt on all input?
   b) Does the given Turing machine halt on no input?
   c) Does the given Turing machine halt on at least one input?
   c) Does the given Turing machine fail to halt on at least one input?

3. Can you construct a Turing machine, which gets as it input a pair $c_M w$ and decides with less than K transitions, whether M halts with at most K transitions? (K is some constant.) I.e. your machine should recognize language $L = \{c_M w |$ Computation of M with input w takes $\leq K$ transitions.$\}$ and in addition your solution machine takes itself $< K$ transitions. (Hint: Contradiction method!)

4. Invent some concrete examples of unsolvable problems! (Other than halting problem.) N.B.! Justify the unsolvability. Is the problem partially solvable (i.e. recursive enumerable language) or totally unsolvable problem?

5. Are the following problems solvable or unsolvable? If they are unsolvable, tell if they are partially solvable or totally unsolvable. If they are solvable, give the weakest type of machine (finite automaton, pushdown automaton, Turing machine), which solves the problem!

   a) Give the integer solution of x for arbitrary polynom $P(x) = a_1 x^n + a_2 x^{n-1} + ... + a_1 x + a_0$! (i.e. such x that $P(x) = 0$)

   b) Program for compiler an a error checker, which gets as its input a code of a program and checks that the program doesn't perform division by 0 during it execution.

   c) Seach if the given text file contains either words **cat** and **black** or words **bird** and **white**, but not both.

d) Check that in the given text file word "animal" is followed by word "wise", only if word "cat" appears in the same sentence.

e) Construct a general virus tester, which recognizes all "dangerous" programs i.e. such programs, which can change system files during their execution.

6. Describe some problem briefly, like in the previous task. Write your problem description on a piece of paper so that it can be used in the following game. Try to analyze the solvability and difficulty of your problem!

7. Participate the problem classification game! 3 exercise points for all participants.

More challenging:

8. Let L be a recursive enumerable language and $\overline{L}$ non-recursive enumerable language. Is language L',

   $L' = \{0w|w \in L\} \cup \{1w|w \notin L\}$,

   recursive, recursive enumerable or non-recursive enumerable? Justify your answer carefully!

9. The following problems are known to be unsolvable, but are they partially solvable or totally unsolvable?

   a) Does the language $L(M)$ contain at least two strings?
   b) Is $L(M)$ finite?
   c) Is $L(M)$ context-free language?

10. We know that for recursive reduction of problems $\leq_m$ holds:

    i) If $A \leq_m B$ and B is recursive enumerable, then A is recursive enumerable.

    i) If $A \leq_m B$ and B is recursive, then A is recursive.

    Is language X recursive, recursive enumerable or totally unsolvable in the following cases?

    a) For universal language U holds $U \leq_m X$.
    b) For language H and for an unknown recursive enumerable language Y holds $H \leq_m Y$ and $Y \leq_m X$. ($H = \{c_M w|M$ halts on input $w\}$)
    c) For H's complement $\overline{H}$ holds $\overline{H} \leq_m X$.
    d) For Hamiltonian Cycle -language $HC = \{x|x$ codes graph G, which contains

Hamiltonian cycle$\}$ [1] holds $X \leq_m HC$.
e) $HC \leq_m X$

11. Answer the course evaluation query! The course evaluation form can be found on `http://cs.joensuu.fi/ãrvio/english.html`. Tell if you participated problem-based learning and how you felt it! Tell also, if you didn't take the problem-based way, and why it didn't suit for you! Thank you for your feedback!

[1] Hamiltonian cycle=cycle, which goes through all vertices exactly once.