University of Twente
Department of Computer Science,
Faculty EEMCS
October 28, 2010, 8:45 – 12:15h.

Year 2010-2011
Period 1A
Interim examination,

# Computability and computational complexity
## (192111700)

The use of course material during the examination is permitted. To hand in work produced with the help of others (by cribbing, by passing notes or other messages, by wireless consultation of persons or (internet) sources during the examination), is fraudulent.

Your answers to questions must be supported by a clear argument. To give an answer without explanation will have a negative influence on the number of points you will earn. This applies also to incomplete and unclear explanations.

You can earn 90 points in total. To determine your mark, the number of points earned is incremented by 10, and the total is divided by 10. The result is rounded to the nearest natural number.

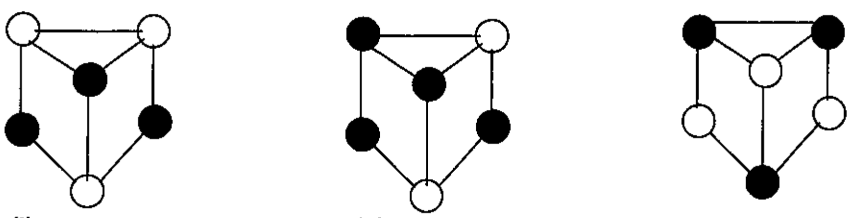
There are 6 exercises, on 3 pages. The exercises are **not** ordered in increasing difficulty (nor conversely). Moreover, the number of points to be earned for an exercise is not necessarily a measure for the difficulty of that exercise.

Please put your name, your student number and the course name on every sheet of paper you hand in.

Veel succes, good luck, bonne chance!!

**Exercise 1** (12 points).

The Vertex Cover problem is as follows. Given a number *k* and a graph G=⟨V,E⟩, V the set of vertices, E the set of edges, find a vertex cover for G with *k* (or less) elements. A vertex cover is a subset W'⊆ V such that every edge has its source or its target vertex (or both) in W.
The three pictures below show:
(i) three nodes in a graph which do not constitute a vertex cover (the top edge is not covered),
(ii) four vertices in the same graph which do constitute a vertex cover,
(iii) three vertices in the same graph which constitute a vertex cover.



**(i)** **(ii)** **(iii)**

**(a)** Give a representation of the Vertex Cover problem. Use the alphabet $\Sigma = \{0,1,\#\}$.
**(b)** Design a non-deterministic Turing machine which solves the Vertex Cover problem in polynomial time (using the representation you gave in 1a). Do not forget to explain why your solution is indeed a polynomial time solution!

**Exercise 2** (20 points).

Discuss each of the following questions. Do not forget to explain your answers.

**(a)** Let $L \subseteq \Sigma^*$ be a recursive language. Is $L' = \{ v^n \mid v \in L, n > 0 \} \subseteq \Sigma^*$ recursive?

**(b)** Let $\Sigma=\{a,b\}$, and $L \subseteq \Sigma^+$ be the language characterized as follows:

$v \in L$ if and only if the number of *a*'s in *v* the square of the number of *b*'s.

So e.g. *ab*, *abaaba,* and *bbaaaa* are in L, but *aaaaaaaabbb* is not.

Is L a context-sensitive language?

**(c)** Let L be an RE (recursively enumerable) language. Is there a type 0 grammar G such that $L(G) = \{ w \mid w \notin L \}$ (the complement of L).

**(d)** What is the difference between being hard for and being complete for a set C of problems?

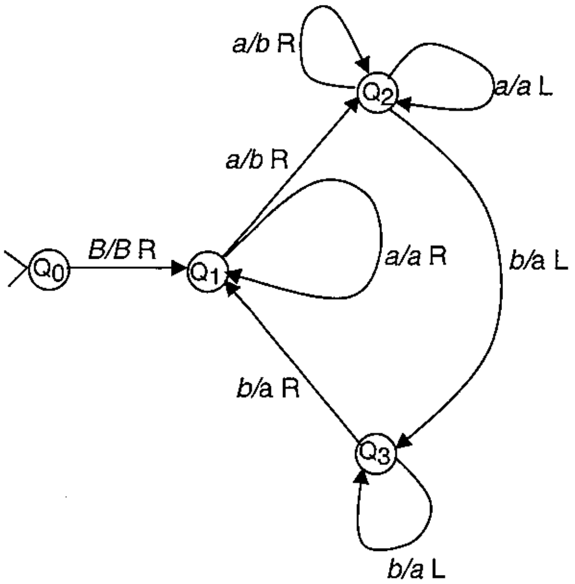
**(e)** Is it decidable whether the intersection of two recursive languages is empty?

**Exercise 3** (12 points).

The picture below shows a non-deterministic TM with input alphabet $\{a\}$ and tape alphabet $\{a,b,B\}$.

**(a)** Give the sequence of configurations which is longest computation leading from $BbQ_2aaaaB$ to $BaQ_1aaaaB$.

**(b)** Is $tc_M \in O(n)$? If your answer is yes, give a clear and convincing argument. If it is no, give the best upper bound you can find for $tc_M$.



**Exercise 4** (21 points).

Let M be a Turing machine with input alphabet {1} and define the language $L^M$ by
$L^M =_{def} \{1^k \mid M \text{ with input } 1^k$ does **not** terminate within $2^k$ steps$\}$.

**(a)** Given M, design a multi-tape Turing machine M′ which accepts $L^M$.

**(b)** Is $L^M$ a recursive language?

**(c)** Is the language $\{R(M) \mid L^M \neq \varnothing\}$ recursively enumerable?

**Exercise 5** (12 points).

Let $L \subset \{0,1\}^*$ be the following language:

L={ R(M)$w$ | M with input $w$ has a computation which terminates within $|w|^2$ steps}.

The M's in this definition are non-deterministic TM's, and $|w|$ denotes the length of $w$.

**(a)** Is $L \in$ NP?

**(b)** What can you say about $L \in$ P?

**Exercise 6** (13 points).

We consider multi-tape TM's with the "no-overwrite" property. "No overwrite" means that the possible transitions of the machine are limited by the following constraint: contents of cells which contain a non blank tape symbol can not be erased or changed.
Now let L be the language $L = \{0^i 1^{2i} 0^i \mid i>0\} \subset \{0,1\}^*$

**(a)** Design a multi-tape TM with the no overwrite property which accepts L, with space bound $2n$. Carefully explain the working of your design, and the argument for the space bound.