A&C 5: Variants of Turing Machines: non-determinism, oracles.

Theory.

T5.1 Non-deterministic Turing machine,

T5.2 Languages accepted by NDTMs,

T5.3 oracle machines,

T5.4 oracle equivalence.

Exercises.

E5.1 What are the complements of languages accepted by NDTMs?

E5.2 How to simulate a NDTM on a DTM?

E5.3 Construct a NDTM, accepting the following language L_h (assume some reasonable encoding of numbers): L_h consists of sequences S of integers, for which there exists $S' \subseteq S$, such that $\sum_{s \in S'} = \sum_{s \in S \setminus S'}$.

In the following we assume that we have some reasonable encoding of a graph.

E5.4 Construct a NDTM, accepting the following language L_c: L_c consists of pairs (G, k), such that G contains a clique of at least k vertices.

E5.5 Construct a NDTM, accepting the following language L_p: L_p consists of pairs (G, k), such that G contains a simple path with at least k vertices.

E5.6 Construct a NDTM, which computes the longest path in an input graph G.

By RE we denote the set recursively enumerable languages and by REC we denote the set of recursive languages. For a language A, by REC^A we denote the set of languages accepted by oracle machines with oracle A and stop property. For a set S of languages, by $\overline{S}$ we denote the set of complements of languages in S.

E5.7 Show that for every A it holds that $REC^A = \overline{REC^A}$.

E5.8 Show that for every A it holds that $RE^A \cap \overline{RE^A} = REC^A$.

E5.9 Let A be recursive. What is RE^A and REC^A?

E5.10 Construct an oracle machine with worst-case time complexity polynomial, with an oracle L_c, accepting the following language L_i: L_i consists of pairs (G, k), such that G contains an independent set of at least k vertices.

E5.11 Construct an oracle machine with stop property, accepting the diagonal language, using the universal language as an oracle.

E5.12 Construct an oracle machine solving the halting problem, using the universal language as an oracle.