Pina Kolling
piko0011

Summary: Lecture 7

Summary for the chapters *9.1* and *9.2*. [1]

NP-Completeness

NP

Class of lanugages decided by nonderteministic Turing machines in polynomial time.

NP-completeness:

- easiest problems among those we do not know how to solve efficiently
- if P≠NP can be proven: exact border of efficient solvability is found
- best bet for proving P=NP: show that some NP-complete problem is P
- Until then, the NP-complete problems are the least likely ones in NP to be efficiently solved

Lanugage L

$L = \{x : (x, y) \in R \textit{ for some } y\}$
L gets an input x and finds a y with $((x, y) \in R$ and the relation $R \subseteq \Sigma^* \times \Sigma^*$.

Polynomially decidable:

- R is polynomially decidable if there is da deterministic Turing machine deciding the language L in polynomial time
- then the relation R (not the language L) is polynomially decidable

Polynomially balanced:

- R is polynomially balanced if $(x, y) \in R$ implies $|y| \leq |x|^k$ for some $k \geq 1$
  → length of the second component is bounded by a polynomial in the length of the first
- then the relation R (not the language L) is polynomially balanced

TODO
Questions:

Typical problems in NP

Title

Content

TODO
Questions:

3Sat is NP-complete

Title

Content

TODO
Questions:

2Sat in P (graph construction)

Title

Content

TODO
Questions:

2Sat in NL

Title

Content

TODO
Questions:

MaxSat is NP-complete

Title

Content

TODO
Questions:

NaeSat is NP-complete

Title

Content

TODO
Questions:

References

[1] Christos H. Papadimitriou. *Computational Complexity*. Addison-Wesley Publishing Company, 1994.