René Maseli
Prof. Dr. Roland Meyer

Theoretical Computer Science 2
Exercise Sheet 5

TU Braunschweig
Summer semester 2024

Release: 06/10/2024 Due: 06/20/2024, 18:30

(Changes from 06/17/2024 are coloured red, including the addendum on page 3.)

Hand in your answers to the Vips directory of the Stud.IP course until wednesday, 20.06.2024 11:59 pm. You should provide your answers either directly as PDF file or as a readable scan or photo of your handwritten notes. Submit your results as a group of four. On the front page, state the **degree programme, name, surname and student id** of each member of your group.

Homework Exercise 1: Some graph problems are NL-complete… [8 points]

In the lecture you were shown that PATH is NL-complete. The following problems related to PATH are also NL-complete.

Accessibility with intermediate node (INTERPATH)

Given:	Directed, acyclic graph $G = \langle V, \rightarrow \rangle$, Vertices $s, t, u \in V$
Question:	Is there a path in G starting in s, running through t and ending in u?

a) [4 points] Show that IREACH is NL-complete wrt. LogSpace reductions, by first proving INTERPATH $\leq_m^{\log}$ PATH, and then PATH $\leq_m^{\log}$ INTERPATH.

Acyclicity (ACYC)

Given:	Directed graph $G = \langle V, \rightarrow \rangle$
Question:	Is there no cycle in G?

b) [4 points] For the problem ACYCPATH, we assumed without checking, that the input graph is acyclic. Now show, that that check for this property, ACYC, is already an NL-complete problem (wrt. LogSpace-reductions).

Homework Exercise 2: Integer Programming [6 points]

Consider the following arithmetic problem.

Integer Programming$_2$ (IP$_2$)

Given:	$m, n \in \mathbb{N}$, Matrix $A \in \mathbb{Z}^{m \times n}$, Vector $b \in \mathbb{Z}^m$,
	where all rows of A have at most two non-zero values.
Question:	Gibt es **kein** $x \in \{0, 1\}^n$ mit $Ax \geq b$?

a) [4 points] Show IP$_2$ $\leq_m^{\log}$ 2SAT, and consequentially, that IP$_2$ $\in$ NL holds.

Hint: $Ax \geq b$ means, that for all rows $i \leq m$, the inequality $A_i \cdot x \geq b_i$ holds. Utilize, that + and $\geq$ are LogSpace-computable.

b) [2 points] Show that 2SAT $\leq_m^{\log}$ IP$_2$ holds, and with this, that IP$_2$ is NL-hard resp. LogSpace-many-one reductions.

Homework Exercise 3: Completeness in L [5 points]

Prove:

a) [4 points] Let $B \in \mathrm{L}$ be non-trivial and A be an arbitrary problem. We can show $A \in \mathrm{L}$ if, and only if, $A \leq_m^{\log} B$.

b) [1 point] Every non-trivial problem $A \in \mathrm{L}$ is already L-complete with respect to LogSpace-many-one reductions.

Exercise 4:

Enrich your collection of NL-complete problems.

Non-Emptiness of regular languages (NONEMPTY-REG)	
Given:	A Turing machine M.
Question:	Are M regular and $\mathcal{L}(M) \neq \emptyset$?

a) Show NONEMPTY-REG ∈ NL by describing the workings of a suitable nondeterministic decider with logarithmic-bounded space complexity.
Hint: You may assume, that 'M is regular' is deterministicly logspace-computable.

b) Show that NONEMPTY-REG is NL-hard with respect to logspace many-one reductions by giving a reduction for PATH $\leq_m^{\log}$ NONEMPTY-REG.

Infinitiy of regular languages (INF-REG)	
Given:	A Turing Machine M.
Question:	Are M regular and $\mathcal{L}(M)$ infinite?

c) Show that INF-REG is NL-complete wrt. logspace many-one reductions.

Exercise 5:

Prove the following lemmas:

a) Let $f, g : \mathbb{N} \to \mathbb{N}$ be two functions and $m \geq m' \in \mathbb{N}$ be numbers of tapes. If $\forall x \in \mathbb{N} : g(x) \leq f(x)$ and $\mathrm{NTIME}_m(f) \subseteq \mathrm{DTIME}_{m'}(g)$ hold, then we get $\mathrm{NTIME}_m(f) = \mathrm{coNTIME}_m(f)$.

b) Let $\mathcal{C}$ be a complexity class, R be a set of functions and $A \in \mathcal{C}$ a problem. If A is $\mathcal{C}$-hard/complete w.r.t R-many-one-reductions, then $\overline{A}$ is co$\mathcal{C}$-hard/complete w.r.t R-many-one-reductions.

Exercise 6:

Im folgenden betrachten wir die Klassen der NL und NL-vollständigen Probleme.

a) Zeigen Sie, dass die Klasse NL unter Vereinigung, Durchschnitt, Komplement und Kleene-Stern abgeschlossen ist.

b) Nun untersuchen Sie die Klasse der NL-harten Probleme auf Abgeschlossenheit unter diesen Operationen.

Because it did not make it into the tutorium:

Lemma: PATH $\leq_m^{\log}$ ACYCPATH.

Beweis:
We need f : Instances(PATH) $\rightarrow$ Instances(ACYCPATH)
satisfying $\langle G, s, t\rangle \in$ PATH $\iff f(G, s, t) \in$ ACYCPATH.
Idea: The vertices shall know their own distance from s.
To accomplish this, create $|V|$ copies of each vertex (maximal path length): $V' = V \times \{0, \ldots, |V|\}$.
Every edge increases the distance: $\forall u \rightarrow v \land 0 \leq i < |V| : \langle u, i\rangle \rightarrow' \langle v, i+1\rangle$.
The actual length of a path to t is irrelevant: $\forall 0 \leq i < |V| : \langle t, i\rangle \rightarrow' \langle t, i+1\rangle$.
f shall compute $\langle G', \langle s, 0\rangle, \langle t, |V|\rangle\rangle$ with $G' = \langle V', \rightarrow'\rangle$.

LogSpace-computable: Iterate over all i and print the slightly-modified edges into the output tape. In the worst case, the number of vertices and edges gets squared.

Sound: The modified graph is always acyclic, since every potential cycle had to contain at least one edge with equal or descending i-component $\langle x, i+j\rangle \rightarrow' \langle y, i\rangle$, which cannot exist by construction.

Hence $\langle G, s, t\rangle \in$ PATH	$\iff$	$\exists$ path of length $k \leq |V|$ in G from s to t	Def. PATH
	$\iff$	$\exists$ path in G' from $\langle s, 0\rangle$ to $\langle t, k\rangle$	Construction
	$\iff$	$\exists$ path in G' from $\langle s, 0\rangle$ to $\langle t, |V|\rangle$	Construction
	$\iff$	$\langle G', \langle s, 0\rangle, \langle t, |V|\rangle\rangle \in$ ACYCPATH	Def. ACYCPATH .

□