Exam Theoretical Computer Science - Bridging Course

Tuesday, February 21, 2017, 9:00-10:30

Name: 
Matriculation Nr.: 
Signature: 

Do not open or turn until told so by the supervisor!

- Write your name and matriculation number on this page and sign the document!
- Write your name on all sheets!
- Your signature confirms that you have answered all exam questions without any help, and that you have notified exam supervision of any interference.
- Write legibly and only use a pen (ink or ball point). Do not use red! Do not use a pencil!
- This is an open book exam therefore printed or hand-written material is allowed.
- However, no electronic devices are allowed.
- There are seven tasks (with several sub-tasks each) and there is a total of 90 points.
- 40 points are sufficient in order to pass the exam.
- Only one solution per task is considered! Make sure to strike out alternative solutions, otherwise the one yielding the minimal number of points is considered.
- Detailed steps might help you to get more points in case your final result is incorrect.
- The keywords Show... or Prove... indicate that you need to prove or explain your answer carefully.
- The keywords Give... or State... indicate that you only need to provide a plain answer.
- Read each task thoroughly and make sure you understood what is required of you.

<table>
<thead>
<tr>
<th>Task</th>
<th>Max. Points</th>
<th>Achieved</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>14</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>13</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>Σ</td>
<td>90</td>
<td></td>
</tr>
</tbody>
</table>
Task 1: Regular Languages (11 Points)

Let $L = \{ab(ab)^n b^m \mid n, m \in \mathbb{N} \cup \{0\}\}$ be a language over the alphabet $\Sigma = \{a, b\}$.

(a) Give a regular expression that generates $L$ (1 point).

(b) Give a nondeterministic finite automaton that recognizes $L$ and has at most three states (5 points).

(c) Give a deterministic finite automaton that recognizes $L$ (5 points).

Remark: You can directly give a deterministic automaton. No intermediate steps required.
Task 2: Context-Free Languages (14 Points)

Consider the language \( L = \{(ab)^n(cd)^n \mid n \in \mathbb{N}\} \) over alphabet \( \Sigma = \{a, b, c, d\} \). Note that 0 \( \notin \mathbb{N} \).

(a) Give a context-free grammar that generates \( L \) (2 points).

(b) Give a grammar in Chomsky Normal Form (CNF) that generates \( L \) (5 points).
   
   **Remark:** You can directly give a grammar in CNF for \( L \), no intermediate steps required.

(c) Prove that \( L \) is not a regular language. Use the Pumping Lemma (7 points).
Task 3: Turing Machines

Consider the Turing machine \( M \) over the alphabet \( \Sigma = \{a, b\} \), which is given via the following state diagram. Note: The blank symbol \( \square \in \Gamma \) represents an empty cell on the tape.

(a) Simulate \( M \) with input \( s_1 = ab \) on its tape until it halts. Give all configurations that \( M \) passes through. State whether \( s_1 \in L \) or not (3 points).

(b) Simulate \( M \) with input \( s_2 = abb \) on its tape until it halts. Give all configurations that \( M \) passes through. State whether \( s_2 \in L \) or not (4 points).

(c) Give a description of the language \( L(M) \) that \( M \) recognizes in the form of a set (3 points).
Task 4: \( O \) - Notation (13 Points)

State whether the following claims are true or false (1 point each). Then prove or disprove the claim (5 points for (a) and 6 points for (b)). Use the definition of the \( O \)-notation.

(a) \( \log_2(2^n \cdot n^3) \in O(n) \).

(b) \( 3\sqrt{n^2} \in O(\sqrt{n}) \).

Hint: You may use that \( \log_2 n \leq n \) for all \( n \in \mathbb{N} \).
Task 5: Decidability

(a) Consider the problem VERTEXCOVER:

\[
\text{VERTEXCOVER} := \{ \langle G, k \rangle \mid \text{undirected, simple graph } G \text{ has a vertex cover of size } k \}.
\]

A vertex cover of size \(k\) of \(G = (V, E)\) is a subset \(C \subseteq V\) of nodes, such that \(|C| = k\) and for all \(\{u, v\} \in E\) it holds that \(u \in C\) or \(v \in C\).

Show that VERTEXCOVER is decidable by giving an algorithm (abstract description or pseudo-code) that decides whether a graph has a vertex cover of size \(k\) or not (6 points).

Explain why your algorithm accepts exactly the instances \(\langle G, k \rangle\) which have a vertex cover of size \(k\) (1 point) and why it always halts (1 point).

(b) Let \(c \in \mathbb{N}\) be a fixed constant and let \(\Sigma\) be a finite alphabet.

Consider the length-restricted Halting problem defined over \(\Sigma\).

\[
H_c := \{ \langle M, s \rangle \mid \text{Turing machine } M \text{ halts on input } s \text{ and } |\langle M, s \rangle| \leq c \},
\]

\(H_c\) is similar to the usual Halting problem but restricted to strings \(\langle M, s \rangle\) shorter than \(c\).

State whether \(H_c\) is decidable or not (2 points). Proof your claim (5 points).
Consider the following problems

**INDEPENDENTSET**: \( \{ \langle G, k \rangle \mid \text{undirected, simple graph } G \text{ has an independent set of size } k \} \).

An **independent set** of size \( k \) of \( G = (V, E) \) is a subset \( I \subseteq V \) of nodes, such that \( |I| = k \) and \( \{u, v\} \notin E \) for all \( u, v \in I, u \neq v \).

**CLIQUE**: \( \{ \langle G, k \rangle \mid \text{undirected, simple graph } G \text{ has a clique of size } k \} \).

A **clique** of size \( k \) of \( G = (V, E) \) is a subset \( C \subseteq V \) of nodes, such that \( |C| = k \) and \( \{u, v\} \in E \) for all \( u, v \in C, u \neq v \).

Use that **CLIQUE** is a known \( \mathcal{NP} \)-complete problem to **show** that **INDEPENDENTSET** is \( \mathcal{NP} \)-complete.

**Remark**: Document the steps of your proof carefully.
Task 7: Logic  

(a) Consider the following **propositional logical** entailment

\[ \{ p \lor r \lor s, p \lor \neg r, \neg p \lor q, \neg q \} \models s \land \neg r \]

(i) Use known equivalencies to convert the above entailment into the form \( KB \models \bot \), where \( KB \) is in **Conjunctive Normal Form** (3 points).

(ii) Use the **Resolution Calculus** to prove this logical entailment (5 points).

(b) Consider the following **first order logical** formula

\[ \varphi := [\forall x \forall y f(s(x, y)) = s(f(x), f(y))] \land [\exists x \exists y \neg (f(x) = f(y))] \]

where \( x, y \) are variable symbols and \( f, s \) are function symbols.

(i) Give an interpretation which is **no model** of \( \varphi \) (3 points).

(ii) Give a **model** of \( \varphi \) (4 points).

Remark: No proof required. Mark which interpretation is a model and which is not.