Johannes Blömer
Jan Bobolz, Gennadij Liske 14. April 2016

Complexity Theory

SS 2016

Class Handout 1

Exercise 1:
Consider the DTM M $= (Q, \Sigma, \Gamma, \delta, q_0, q_{\text{reject}}, q_{\text{reject}})$ with $Q = \{q_0, q_1, q_2, q_3, q_{\text{reject}}, q_{\text{reject}}\}$, $\Sigma = \{0, 1\}$, $\Gamma = \Sigma \cup \{\triangleright, \sqcup\}$ and transition function δ defined by

$$\begin{array}{rll} \delta: & (q_0, \triangleright) \mapsto (q_0, \triangleright, R), & (q_1, \triangleright) \mapsto (q_1, \triangleright, R), \\ & (q_0, 0) \mapsto (q_1, \sqcup, R), & (q_1, 0) \mapsto (q_1, 0, R), \\ & (q_0, 1) \mapsto (q_{\text{reject}}, 1, R), & (q_1, 1) \mapsto (q_1, 1, R), \\ & (q_0, \sqcup) \mapsto (q_{\text{reject}}, \sqcup, R), & (q_1, \sqcup) \mapsto (q_2, \sqcup, L), \\ \\ & (q_2, \triangleright) \mapsto (q_2, \triangleright, R), & (q_3, \triangleright) \mapsto (q_3, \triangleright, R), \\ & (q_2, 0) \mapsto (q_{\text{reject}}, 0, R), & (q_3, 0) \mapsto (q_3, 0, L), \\ & (q_2, 1) \mapsto (q_3, \sqcup, L), & (q_3, 1) \mapsto (q_3, 1, L), \\ & (q_2, \sqcup) \mapsto (q_2, \sqcup, L), & (q_3, \sqcup) \mapsto (q_0, \sqcup, R). \end{array}$$

Determine the consecutive configurations of the DTM M on input $w_1 = 010101$ as well as on input $w_2 = 000111$. Which language does the DTM M decide?

Exercise 2:
Consider the following language

$$L = \left\{ 0^{2^n} \;\middle|\; n \in \mathbb{N} \right\} \quad .$$

Construct a DTM M which decides L. Show that $L \in \mathbf{P}$.

Exercise 3:
Consider the following complexity class

$$\mathbf{DTIMESPACE}\big(t(n), s(n)\big) = \left\{ L \subseteq \{0,1\}^* \;\middle|\; \begin{array}{l} \text{There exists a DTM which decides } L \\ \text{in time } \mathcal{O}\big(t(n)\big) \text{ and in space } \mathcal{O}\big(s(n)\big). \end{array} \right\} \quad .$$

What is the difference to the class $\mathbf{DTIME}\big(t(n)\big) \cap \mathbf{DSPACE}\big(s(n)\big)$? Which inclusion relations exist between these classes?

Exercise 4:
Consider the BFS algorithm and analyze its space complexity.

Algorithm 1: Breadth-First-Search:

```
Input   : (G = (V, E), root ∈ V)
foreach v ∈ V ;                                        // initialization
 do
   v.distance := ∞;
   v.parent := null;
end
Create empty queue Q;
Q.enqueue(root);
/* Compute the distance and a shortest path from root to every node.   */
root.distance := 0;
while Q is not empty: do
   current := Q.dequeue();
   foreach v ∈ V with (current, v) ∈ E: do
      if v.distance == ∞ then
         v.distance := current.distance + 1;
         v.parent := current;
         Q.enqueue(u);
      end
   end
end
```

Exercise 5:
Explain:

- $L \in \mathbf{P}$ implies $\overline{L} \in \mathbf{P}$.
- $\mathbf{P} \subseteq \mathbf{PSPACE}$.
- For all $k \in \mathbb{N}$ it holds $\mathbf{DSPACE}_k\big(s(n)\big) = \mathbf{DSPACE}\big(s(n)\big)$.
- SAT $\in \mathbf{PSPACE}$.