Exercise 1: Decidability

Let $\Sigma$ be a fixed finite alphabet. Show that the language of deterministic finite automata (DFAs) on $\Sigma$ that accept every word is decidable. Formally, show that

$$L = \{ \langle A \rangle \mid A \text{ is a deterministic finite automaton with } L(A) = \Sigma^* \}$$

is a decidable language.

Remark: You can use that it is not difficult to construct a Turing machine which tests whether an input is the well formed encoding of a deterministic finite automaton.

Sample Solution

Let $B$ be a DFA such that the language generated by $B$ is $\Sigma^*$. That is, $L_B = \Sigma^*$. (It is easy to see that this is always possible for any given alphabet $\Sigma$.) We have shown in the video lecture that testing equivalence for two DFA is a decidable problem. Let $M$ be such a turing machine that can test equivalence for two DFA. We can construct a turing machine $M'$, such that upon input $\langle A, B \rangle$ where $A$ is a DFA, it will run $M$ with input $\langle A, B \rangle$. If $M$ accepts the input, then $M'$ accepts the input $A$ as well, otherwise $M'$ rejects. Since $M$ will give definite answer in finite time, we know $M'$ will give definite answer in finite time as well. Hence, we know $L$ is decidable.

Exercise 2: Landau Notation

The set $O(f)$ contains all functions that are asymptotically not growing faster than the function $f$ (when additive or multiplicative constants are neglected). That is:

$$g \in O(f) \iff \exists c \geq 0, \exists M \in \mathbb{N}, \forall n \geq M : g(n) \leq c \cdot f(n)$$

For the following pairs of functions, check whether $f \in O(g)$ or $g \in O(f)$ or both. Prove your claims (you do not have to prove a negative result $\notin$, though).

(a) $f(n) = 100n$, $g(n) = 0.1 \cdot n^2$
(b) $f(n) = \sqrt{n^3}$, $g(n) = \sqrt{n}$
(c) $f(n) = \log_2(2^n \cdot n^3)$, $g(n) = 3n$

Hint: You may use that $\log_2 n \leq n$ for all $n \in \mathbb{N}$. 
Sample Solution

(a) It is $100n \in \mathcal{O}(0.1n^2)$. To show that we require constants $c, M$ such that $100n \leq c \cdot 0.1n^2$ for all $n \geq M$. Obviously this is the case for $c = 1000$ and $M = 1$.

(b) We have $g(n) \in \mathcal{O}(f(n))$. Let $c := 1$ and $M := 1$. Then we have

\[
g(n) \leq c \cdot f(n) \quad (1) \\
\Leftrightarrow \quad \sqrt{n} \leq n^{2/3} \quad (2) \\
\Leftrightarrow \quad 1 \leq n^{1/6} \quad (3) \\
\Leftrightarrow \quad 1 \leq n \quad (4)
\]

The last inequality is satisfied because $n \geq M = 1$.

(c) $f(n) \in \mathcal{O}(g(n))$ holds. We give $c > 0$ and $M \in \mathbb{N}$ such that for all $n \geq M : \log_2(2^n \cdot n^3) \leq c \cdot n$. As $c_1n \in \mathcal{O}(g(n))$ holds for every constant $c > 0$ the result will follow with the transitivity of the $\mathcal{O}$-notation.

\[
\log_2(2^n \cdot n^3) = \log_2(2^n) + \log_2(n^3) = n + 3 \cdot \log_2(n) \leq n + 3n = 4n.
\]

Thus $\log_2(2^n \cdot n^3) \leq c \cdot n$ for $n \geq M := 1$ and $c := 4$.

We also have that $g(n) \in \mathcal{O}(f(n))$ holds because

\[
g(n) = 3n \leq 3(n + 3 \cdot \log_2(n)) = 3(\log_2(2^n \cdot n^3)) = 3 \cdot f(n).
\]

Thus with $c = 3$ and for $n \geq M := 1$ we have $g(n) \leq cf(n)$.

Exercise 3: Sorting Functions by Asymptotic Growth \hspace{1cm} (6 Points)

Sort the following functions by asymptotic growth using the $\mathcal{O}$-notation. Write $g \prec \mathcal{O} f$ if $g \in \mathcal{O}(f)$ and $f \not\in \mathcal{O}(g)$. Write $g = \mathcal{O} f$ if $f \in \mathcal{O}(g)$ and $g \in \mathcal{O}(f)$.

<table>
<thead>
<tr>
<th>Function</th>
<th>$n^2$</th>
<th>$\sqrt{n}$</th>
<th>$3^n$</th>
<th>$n^{100}$</th>
<th>$2^n$</th>
<th>$\log(n)$</th>
<th>$n!$</th>
<th>$\log(n^2)$</th>
<th>$(\log n)^2$</th>
<th>$n \log n$</th>
<th>$10^{100} n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\prec \mathcal{O}$</td>
<td>$\sqrt{\log n}$</td>
<td>$\log(\sqrt{n})$</td>
<td>$\log n$</td>
<td>$n$</td>
<td>$\log(n^2)$</td>
<td>(10^{100} n)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\prec \mathcal{O}$</td>
<td>$n \log n$</td>
<td>$n^2$</td>
<td>$n^{100}$</td>
<td>$2^n$</td>
<td>$n!$</td>
<td>$n^n$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\prec \mathcal{O}$</td>
<td>$n \cdot 2^n$</td>
<td>$3^n$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Sample Solution

<table>
<thead>
<tr>
<th>Function</th>
<th>$\sqrt{\log n}$</th>
<th>$\log(\sqrt{n})$</th>
<th>$\log n$</th>
<th>$n$</th>
<th>$\log(n^2)$</th>
<th>(10^{100} n)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\prec \mathcal{O}$</td>
<td>$\log(n^2)$</td>
<td>(10^{100} n)</td>
<td>$n^2$</td>
<td>$n^{100}$</td>
<td>$2^n$</td>
<td>$n^n$</td>
</tr>
<tr>
<td>$\prec \mathcal{O}$</td>
<td>$n \log n$</td>
<td>$n^2$</td>
<td>$n^{100}$</td>
<td>$2^n$</td>
<td>$n!$</td>
<td>$n^n$</td>
</tr>
</tbody>
</table>