Dominik Wawszczak
Student ID Number: 440014
Group Number: 1

Languages, Automata and Computation II
Assignment 1

Problem 1

We define an ω-word u as *universal* if, and only if, for any $i \in \mathbb{N}$, there exists a $j > i$ such that $u[i] \neq u[j]$. Intuitively, this means that u contains an infinite subsequence of the form $(ab)^\infty$. Note that if $u \in \Sigma^\omega$ is universal, then for any $v \in \Sigma^\omega$, it holds that $v \sqsubseteq u$. This is because we can remove finitely or infinitely many letters from u to obtain $(ab)^\infty$, and for each i-th pair of consecutive ab occurrences, we can remove either a or b, depending on the value of $v[i]$.

If an ω-word u is not universal, then either $\#_a(u)$ or $\#_b(u)$ is finite. In the first case, u is of the form vb^∞, and we will refer to such an ω-word as *a-long*. In the second case, u is of the form va^∞, and we will call it *b-long*. In both cases, v is a word of finite length.

Let $u_1, u_2, \ldots$ be any infinite sequence of ω-words. We will prove that there exist indices $i < j$ such that $u_i \sqsubseteq u_j$. If there are at least two universal ω-words in this sequence, the proof is straightforward since u_1 can embed into the second universal word in the sequence. Otherwise, either there are infinitely many a-long words in the sequence, or there are infinitely many b-long words. Without loss of generality, assume there are infinitely many a-long words in the sequence. Let $k_1, k_2, \ldots$ be the indices of the a-long words. Denote $u_{k_l} = v_{k_l} a^\infty$, where v_{k_l} is finite. By Higman's lemma, there exist indices $i < j$ such that v_{k_i} is a substring of v_{k_j}, concluding the proof as $u_{k_i} \sqsubseteq u_{k_j}$.

Now, we will show that in the variant where only finitely many letters can be removed, the resulting relation is not a well-quasi-order. Consider the infinite sequence of ω-words $u_1, u_2, \ldots$ such that

$$u_i \;=\; a^i b a^{i+1} b a^{i+2} b \ldots$$

Clearly, for any $i \in \mathbb{N}^+$, we have $u_i \sqsupseteq u_{i+1}$ because the first $i+1$ letters of u_i can be removed to obtain u_{i+1}.

Suppose, for the sake of contradiction, that there exists $i \in \mathbb{N}^+$ such that $u_i \sqsubseteq u_{i+1}$. Let k be the smallest number such that a non expandable block of the form $a^k b$ remains intact, after removing finitely many letters from u_{i+1} to obtain u_i. The number of letters b in u_{i+1} before this block equals $k - i - 1$. However, the number of letters b in u_i before this block equals $k - i > k - i - 1$, which is a contradiction, as letters can only be removed.

From the above, we conclude that $u_i \sqsupset u_{i+1}$ for any $i \in \mathbb{N}^+$. Therefore, $u_1, u_2, \ldots$ forms an infinite sequence of strictly decreasing elements, which shows that the relation is not a well-quasi-order.

Problem 2

Consider the structure $\big((\mathbb{N} \cup \{\omega\})^d, \leqslant\big)$, where $(x_1, \ldots, x_d) \leqslant (y_1, \ldots, y_d)$ if and only if, for each $i \in \{1, \ldots, d\}$, one of the following holds:

$$x_i = y_i = \omega \qquad \text{or} \qquad x_i \in \mathbb{N} \;\wedge\; y_i = \omega \qquad \text{or} \qquad x_i, y_i \in \mathbb{N} \;\wedge\; x_i \leqslant y_i.$$

By Dickson's lemma, this structure is a well-quasi-order because $(\mathbb{N} \cup \{\omega\}, \leqslant)$ is itself a well-quasi-order.

For any tuple $(x_1, \ldots, x_d) \in (\mathbb{N} \cup \{\omega\})^d$, we say $(x_1, \ldots, x_d) \in X$ if and only if, for every $n \in \mathbb{N}$, the following condition holds:

$$\big([x_1 = \omega] \cdot n + [x_1 \neq \omega] \cdot x_1, \ldots, [x_d = \omega] \cdot n + [x_d \neq \omega] \cdot x_1\big) \;\in\; X,$$

where $[P]$ denotes the Iverson bracket, i.e., $[P] = 1$ if P is true, and $[P] = 0$ otherwise.

A subset $A \subseteq \{1, \dots, d\}$ is called *good* if and only if $([1 \in A] \cdot \omega, \dots, [d \in A] \cdot \omega) \in X$. For any good set A, define $g(A)$ as the set of tuples $(x_1, \dots, x_d) \in (\mathbb{N} \cup \{\omega\})^d$ satisfying:

1. $x_i = \omega$, for all $i \in A$;
2. $(x_1, \dots, x_d) \in X$;
3. $(x_1, \dots, x_{i-1}, x_i + 1, x_{i+1}, \dots, x_d) \notin X$, for all $i \in \{1, \dots, d\} \setminus A$.

Suppose, for the sake of contradiction, that $g(A)$ is infinite. Let

$$g(A) \;=\; \{(x_1^1, \dots, x_d^1), (x_1^2, \dots, x_d^2), \dots\}.$$

Since $((\mathbb{N} \cup \{\omega\})^d, \leqslant)$ is a well-quasi-order, there exist indices $i, j \in \mathbb{N}^+$ such that $i < j$ and $(x_1^i, \dots, x_d^i) \leqslant (x_1^j, \dots, x_d^j)$, contradicting the third condition for tuples in $g(A)$.

Define $\mathrm{down}(g(A))$ as the set of all tuples $(y_1, \dots, y_d) \in \mathbb{N}^d$ for which there exists a tuple $(x_1, \dots, x_d) \in g(A)$ such that $(x_1, \dots, x_d) \geqslant (y_1, \dots, y_d)$ and $y_i = 0$ for all $i \in A$. If $g(A)$ is empty, let $\mathrm{down}(g(A))$ be the singleton containing the zero tuple. The set $\mathrm{down}(g(A))$ is finite, with an upper bound on its size:

$$|\mathrm{down}(g(A))| \;\leqslant\; \sum_{(x_1,\dots,x_d) \in g(A)} \; \prod_{i \in \{1,\dots,d\} \setminus A} (x_i + 1).$$

Using this, define a semilinear set

$$Y \;=\; \bigcup_{A \subseteq \{1,\dots,d\} \,\wedge\, A \text{ is good}} \left(\mathrm{down}(g(A)) + \left(\bigcup_{i \in A} ([1 = i], \dots, [d = i]) \right)^{*} \right).$$

By definition of $g(A)$, we have $Y \subseteq X$. Our goal is to prove that $Y = X$.

Suppose, for the sake of contradiction, that there exists a tuple $(x_1, \dots, x_d) \in X \setminus Y$. We will construct a tuple $(y_1, \dots, y_d) \in \mathbb{N}^d$ such that $(x_1, \dots, x_d) \leqslant (y_1, \dots, y_d)$ and $(y_1, \dots, y_d) \in X$. Since Y is downward closed, this will imply $(y_1, \dots, y_d) \in X \setminus Y$.

Proceed through indices $i = 1, \dots, d$, choosing y_i iteratively while maintaining a set A, initially empty. Assume $y_1, \dots, y_{i-1}$ have been chosen. For all $j \in \{1, \dots, i-1\}$, define $y_j' = \omega$ if $j \in A$, and $y_j' = y_j$ otherwise. If there exists a number $z_i \in \mathbb{N}$ such that

$$(y_1', \dots, y_{i-1}', z_i, x_{i+1}, \dots, x_d) \;\in\; X \qquad \text{and} \qquad (y_1', \dots, y_{i-1}', z_i + 1, x_{i+1}, \dots, x_d) \;\notin\; X,$$

set $y_i = z_i$. Otherwise, let $y_i = x_i$ and add i to A. Note that $y_i \geqslant x_i$.

By construction, $(y_1, \dots, y_d) \in X$, and A is good. Furthermore,

$$\forall_{i \in \{1,\dots,d\} \setminus A} \; (y_1', \dots, y_{i-1}', y_i + 1, y_{i+1}', \dots, y_d') \notin X,$$

where y_j' is as defined earlier. Thus,

$$(y_1, \dots, y_d) \;\in\; \mathrm{down}(g(A)) + \left(\bigcup_{i \in A} ([1 = i], \dots, [d = i]) \right)^{*}.$$

This contradiction completes the proof that X is a semilinear set.

Now we proceed to the second part of the problem. Let $T \subseteq \mathbb{Z}^d$ be the set of available transitions. Define X as the downward closure of configurations reachable from x.

Consider a directed graph G where $V(G) = (\mathbb{N} \cup \{\omega\})^d$ and $E(G)$ contains pairs $((y_1, \ldots, y_d), (z_1, \ldots, z_d))$ such that there exists a transition $(t_1, \ldots, t_d) \in T$ satisfying:

$$y_i = z_i = \omega \qquad \text{or} \qquad y_i, z_i \in \mathbb{N} \ \wedge \ y_i + t_i = z_i,$$

for each $i \in \{1, \ldots, d\}$. Run a breadth-first search (BFS) on this graph, starting at vertex x.

Whenever a vertex $(y_1, \ldots, y_d)$ is encountered such that there exists an ancestor $(z_1, \ldots, z_d)$ in the BFS-tree with $(z_1, \ldots, z_d) \leqslant (y_1, \ldots, y_d)$, perform the following:

1. For each $i \in \{1, \ldots, d\}$, set $y'_i = \omega$ if $y_i > z_i$, or $y'_i = y_i$ otherwise;
2. If $(y'_1, \ldots, y'_d)$ has not been visited, add it to the BFS queue and mark it as visited;

Note that $(y'_1, \ldots, y'_d) \geqslant (y_1, \ldots, y_d)$, so we do not need to process the neighbors of $(y_1, \ldots, y_d)$. We do this because if $(y_1, \ldots, y_d)$ and $(z_1, \ldots, z_d)$ belong to X, then $(y'_1, \ldots, y'_d)$ also belongs to X.

Suppose, for the sake of contradiction, that the BFS does not terminate. Then there exists an infinite sequence of vertices:

$$Y_0 \ = \ \big((y_1^1, \ldots, y_d^1), (y_1^2, \ldots, y_d^2), \ldots, \big)$$

where each vertex $(y_1^i, \ldots, y_d^i)$ is the parent of the next. Since $\big((\mathbb{N} \cup \{\omega\})^d, \leqslant\big)$ is a well-quasi-order, there exist indices $i, j \in \mathbb{N}^+$ with $i < j$ such that $\big(y_1^i, \ldots, y_d^i\big) \leqslant \big(y_1^j, \ldots, y_d^j\big)$. Now consider the sequence:

$$Y_1 \ = \ \big((y_1^j, \ldots, y_d^j), (y_1^{j+1}, \ldots, y_d^{j+1}), \ldots, \big).$$

In this sequence, each tuple has at least one element equal to ω. Repeating this argument produces sequences $Y_2, Y_3, \ldots$, where each tuple in Y_k contains at least k elements equal to ω. The existence of Y_{d+1} leads to a contradiction. Therefore, the BFS terminates

Let S be the set of visited vertices. The answer is:

$$X \ = \ \bigcup_{y \in S} \left(\mathrm{down}(\{y\}) + \left(\bigcup_{i \in \{1, \ldots, d\} \ \wedge \ y_i = \omega} ([1 = i], \ldots, [d = i]) \right)^* \right).$$

The algorithm is correct because for every y coverable by x, there exists $y' \in S$ such that $y' \geqslant y$.

Problem 4

This problem was solved in collaboration with Kacper Bal and Mateusz Mroczka.

To begin, we construct an injection $f : \{0, 1\}^* \to \mathbb{N}$, defined as:

$$f(s) \ = \ 2^n + \sum_{i=0}^{n-1} [s[i+1] = 1] \cdot 2^i,$$

where s is a string of length n. Note that every natural number except 0 lies in the image of this function.

We will utilize the following formulas:

$$
\begin{aligned}
\psi_{\text{zero}}(a) &:= \underset{b\in\mathbb{N}}{\forall}\ a+b=b,\\
\psi_{\text{one}}(a) &:= \underset{b\in\mathbb{N}}{\forall}\ a\times b=b,\\
\psi_{\text{even}}(a) &:= \underset{b\in\mathbb{N}}{\exists}\ a=b+b,\\
\psi_{\text{pow2}}(a) &:= \underset{b\in\mathbb{N}}{\forall}\left(\left(\neg\psi_{\text{zero}}(b)\ \wedge\ \underset{c\in\mathbb{N}}{\exists}\ b\times c=a\right)\ \Rightarrow\ (\psi_{\text{even}}(b)\ \vee\ \psi_{\text{one}}(b))\right),\\
\psi_{\leqslant}(a,b) &:= \underset{c\in\mathbb{N}}{\exists}\ a+c=b,\\
\psi_{<}(a,b) &:= \underset{c\in\mathbb{N}}{\exists}\ (a+c=b\ \wedge\ \neg\psi_{\text{zero}}(c)),\\
\psi_{\log}(a,b) &:= \psi_{\text{pow2}}(b)\ \wedge\ \psi_{\leqslant}(b,a)\ \wedge\ \psi_{<}(a,2\times b).
\end{aligned}
$$

Next, we construct a function g that transforms a first-order logic sentence $\varphi(s_1,\dots,s_m)$ over the free monoid $\big(\{0,1\}^*,\cdot,0,1\big)$ into a sentence $\psi(a_1,\dots,a_m)$ over the arithmetic structure $(\mathbb{N},+,\times)$. This transformation ensures that $\varphi(s_1,\dots,s_m)$ is true if and only if $\psi(a_1,\dots,a_m)$ is true.

The function g is defined recursively as follows:

$$
\begin{aligned}
g\left(\underset{s\in\{0,1\}^*}{\forall}\ \varphi'(s,s'_1,\dots,s'_{m'})\right) &:= \underset{a\in\mathbb{N}}{\forall}\ (\neg\psi_{\text{zero}}(a)\ \wedge\ g(\varphi'(s,s'_1,\dots,s'_{m'}))),\\
g\left(\underset{s\in\{0,1\}^*}{\exists}\ \varphi'(s,s'_1,\dots,s'_{m'})\right) &:= \underset{a\in\mathbb{N}}{\exists}\ (\neg\psi_{\text{zero}}(a)\ \wedge\ g(\varphi'(s,s'_1,\dots,s'_{m'}))),\\
g(\varphi'(s'_1,\dots,s'_{m'})\ \wedge\ \varphi''(s''_1,\dots,s''_{m''})) &:= g(\varphi'(s'_1,\dots,s'_{m'}))\ \wedge\ g(\varphi''(s''_1,\dots,s''_{m''})),\\
g(\varphi'(s'_1,\dots,s'_{m'})\ \vee\ \varphi''(s''_1,\dots,s''_{m''})) &:= g(\varphi'(s'_1,\dots,s'_{m'}))\ \vee\ g(\varphi''(s''_1,\dots,s''_{m''})),\\
g(\neg\varphi'(s'_1,\dots,s'_{m'})) &:= \neg g(\varphi'(s'_1,\dots,s'_{m'})).
\end{aligned}
$$

The remaining case involves sentences of the form $s'_1\cdot\ldots\cdot s'_{m'} = s''_1\cdot\ldots\cdot s''_{m''}$. Without loss of generality, we assume these are reduced to $s\cdot t=u$. Let a_s be defined as follows: if s is a variable, a_s is the corresponding variable; if s is a constant, a_s is $f(s)$; and if s is an argument, a_s is the corresponding argument. The values a_t and a_u are defined analogously. Define:

$$
g(s\cdot t=u)\ :=\ \underset{b_t\in\mathbb{N}}{\exists}\ (\psi_{\log}(a_t,b_t)\ \wedge\ b_t\times a_s+a_t=a_u+b_t)\,.
$$

The term b_t represents the most significant bit of a_t. Ideally, we would express the equation as $b_t\times a_s+(a_t-b_t)=a_u$, but since subtraction is not available in this framework, we add b_t to the right-hand side instead.

Finally, for each argument a_i of ψ, we impose the restriction $\neg\psi_{\text{zero}}(a_i)$.

By this construction, $\varphi(s_1,\dots,s_m)$ is true if and only if its transformed counterpart $g(\varphi(s_1,\dots,s_m))=\psi(a_1,\dots,a_m)$ is also true, thus completing the proof.