Technische Universität München **Winter term 2015/16**
I7
Prof. J. Esparza / Dr. D. Suwimonteerabuth / Dr. C. Broadbent **13.11.2015**

Automata and Formal Languages — Homework 6

Due **Friday** 20th November 2015 (TA: Christopher Broadbent)

Please note that questions marked ⋆⋆ are extra-challenging; you should not worry if you are unable to solve them as they are harder than what will be expected in the exam. Only consider them if you have time and are interested!

Exercise 6.1

Construct DFAs and lazy DFAs that recognise the set of strings containing the patterns (i) *mammamia*, and (ii) *abracadabra*.

Exercise 6.2

Two-way finite automata are an extension of lazy automata in which the reading head may not only move right or stay put, but also move left. The tape extends infinitely long to both the left and to the right of the input with all cells empty. A word is accepted if the control state is a final state at the moment the head reaches an empty cell to the right of the input for the first time.

(i) Give a two-way DFA for the language $(0+1)^*1(0+1)^4$. How big is this automaton compared to the smallest ordinary DFA for the same language?

(ii) Show that for every $n \in \mathbb{N}$ there exists a two-way DFA for the language $(0+1)^*1(0+1)^n$ with $O(n)$ states. How many states does the minimal ordinary DFA have for each $n \in \mathbb{N}$?

(iii) Sketch an algorithm for membership of a two-way DFA

(iv) ⋆⋆ Sketch an algorithm for emptiness of two-way DFA

(v) ⋆⋆ Prove that the languages recognized by two-way DFA are regular.

Exercise 6.3

Which of the following relations are regular (with the *lsbf*-encoding)? Justify your answers.

(i) $\{ (n, n^2) \mid n \in \mathbb{N} \}$

(ii) $\{ (n, m) \mid n, m \in \mathbb{N} \text{ with } m \leq \sqrt{n} \}$

(iii) $\{ (n, 5n) \mid n \in \mathbb{N} \}$

(iv) $\{ (n, 7+n) \mid n \in \mathbb{N} \}$

(v) $\{ (n, 7+5n) \mid n \in \mathbb{N} \}$

(vi) $\{ (n, m) \mid n, m \in \mathbb{N} \text{ with } 2n + 3m = 2^k \text{ for some } k \in \mathbb{N} \}$

Exercise 6.4

⋆⋆ Suppose that L is a regular language. Let us write $|w|$ for the length of the word $|w|$. Prove that the language $|L| := \{ \mathit{lsbf}(|w|) \mid w \in L \}$ is regular.

[**Hint**: It follows from the final question of Exercise Sheet 2 that there exist $k_0, k_1, \ldots, k_n$ such that

$$|L| = \{ \mathit{lsbf}(k_0 + \lambda_1 k_1 + \cdots + \lambda_n k_n) \mid \lambda_i \in \mathbb{N} \text{ for each } 1 \leq i \leq n \}$$

.

(i) Show that the relation $R_k := \{ (k, \lambda k) \mid \lambda \in \mathbb{N} \}$ is regular for each $k \in \mathbb{N}$.

(ii) Show also that for any set of numbers $N \subseteq \mathbb{N}$ such that *lsbf*(N) is regular, it is the case that the relation

$$S_N := \{ (r, m) \mid r, m \in \mathbb{N}, \text{ such that } r = m + K \text{ for some } K \in N \}$$

is regular.

(iii) Then put these facts together to prove the result (using induction on n and the *Post* operator from the lectures.)]

Solution 6.2

(i) The following two-way DFA does the job:

$$A := \langle \{ 0, 1 \}, Q, \delta, q_0, \{ p_4 \} \rangle$$

where $Q := \{ q_0, p_0, p_1, p_2, p_3, p_4, q_4, q_3, q_2, q_1 \}$ and

$$\delta(0, q_0) = (q_0, R) \qquad \delta(1, q_0) = (p_0, R) \qquad \delta(\_, p_i) = (p_{i+1}, R) \qquad \delta(\_, q_{i+1}) = (q_i, L) \text{ for each } i \in [0, 3]$$

We will give a proof of correctness for the more general construction in the next part.

The residuals of $L := (0+1)^*1(0+1)^4$ are as follows (for all $k \geq 0$):

$$\begin{aligned} L^{0^k} &= (0+1)^*1(0+1)^4 \\ L^{0^k1} &= (0+1)^*1(0+1)^4 \ + \ (0+1)^4 \\ L^{0^k10} &= (0+1)^*1(0+1)^4 \ + \ (0+1)^3 \\ L^{0^k11} &= (0+1)^*1(0+1)^4 \ + \ (0+1)^4 \ + \ (0+1)^3 \\ &\vdots \end{aligned}$$

In fact the residuals will be all those languages of the form $L + \sum_{i \in N}(0+1)^i$ for a set $N \subseteq [0, 4]$ meaning that there are $2^5 = 32$ residuals. Thus the minimal DFA for L must contain 16 states, more than that of the two-way DFA that we have just given.

We will prove the more general case in the next part.

(ii) Let $L_n := (0+1)^*1(0+1)^n$ (for each $n \in \mathbb{N}$). We claim that each such language is recognised by a 2-way DFA given by:

$$A_n = \langle \{ 0, 1 \}, Q_n, \delta_n, q_0, \{ p_n \} \rangle$$

where $Q_n := \{ q_i \mid i \in [0, n] \} \cup \{ p_i \mid i \in [0, n) \}$ and

$$\delta(0, q_0) = (q_0, R) \qquad \delta(1, q_0) = (p_0, R) \qquad \delta(\_, p_i) = (p_{i+1}, R) \qquad \delta(\_, p_n) = (q_{n-1}, L) \qquad \delta(\_, q_{j+1}) = (q_j, L)$$

for each $i \in [0, n)$ and $j \in [0, n-1)$ where $\_$ ranges over $\{ 0, 1 \}$. We will now prove that A_n recognises the intended language.

Consider a word $w \in \{ 0, 1 \}^*$ that contains R occurrences of the symbol 1. For each r such that $0 \leq r \leq R$, we inductively define the prefix r of w by the following:

$$w_0 = \epsilon \qquad w_{r+1} \text{ is the shortest prefix of } w \text{ ending with 1 of which } w_r \text{ is a prefix}$$

In other words, w_r is the prefix of w that ends in the rth occurrence of 1.

When reading such a word w, the automaton maintains the invariant consisting of the conjunction of all of the following:

- If it is in state p_i, then the head of the automaton will be at a position immediately following a prefix of the form $w_{r+1}(0+1)^i$.
- If it is in state q_i, then (i) the head of the automaton will be at a position immediately following a prefix of the form $w_r(0+1)^i$, *and* (ii) for $b_{i+1}, \ldots, b_n \in \{ 0, 1 \}$ s.t. $w_r b_{i+1} \cdots b_n$ is a prefix of w, it is the case that $w_r b_{i+1} \cdots b_n \notin L_n$

This follows from the fact that the initial position of the automaton satisfies the invariant (the first symbol of the input word is in the position immediately following the prefix $w_0 = \epsilon$ and the initial state is q_0) and it can be checked that each transition preserves the invariant.

It follows that $\mathcal{L}(A_n) \subseteq L_n$ since if the accepting state is p_n is attained in the position immediately after the last symbol in the word, then the invariant implies that the word must be of the from $w_{r+1}(0+1)^n$, i.e. of the form $(0+1)^*1(0+1)^n$.

In order to prove that $L_n \subseteq \mathcal{L}(A_n)$ we prove that $\overline{\mathcal{L}(A_n)} \subseteq \overline{L_n}$. Let $w \in \overline{\mathcal{L}(A_n)}$. Then one of two possibilities must hold: (i) the automaton has reached a non-accepting state with its head just past the end of the word, *or* (ii) the automaton's head never reaches a position just beyond the end of the word.

In the first case, the invariant tells us that indeed $w \notin L_n$. It therefore suffices to rule out the second case. That is we must prove that for every word w the automaton's head eventually reaches the position just beyond its end.

Note that the invariant tells us that the head never moves to the left of the word. Therfore if the head also never moves to the right of the word it must be the case that the automaton has an infinite run on w. By inspection of the transition relation we can also see that an infinite run must reach state q_0 infinitely often.

Let $w'b$ be the prefix of w such that b is the right-most position that the head reaches when in state q_0.

We claim that there is a prefix $w'bb_0$ of w such that the automaton will also reach b' in state q_0, which gives the required contradition (since we assumed b is the right-most such position).

If $b = 0$, then this follows immediately. So consider the case when $b = 1$. Then the automaton will move its head to b_0 and adopt state p_0. By induction on the length of the run, we can then see that in fact w must have a prefix of the form $w'bb_0 \cdots b_n$ such that its head will reach each b_i (for $0 \leq i \leq n$) in state p_i and each b_j (for $0 \leq i < n$) in state q_j. In particular it will reach position b_0 in state q_0, as required. □

So there exists a two-way DFA of linear size (with respect to n) recognising each L_n. We now prove that the minimal (ordinary DFA) has exponential size. Since the minimal DFA has size equal to the number of residuals of L_n, it suffices to prove that L_n has exponentially many residuals.

We claim that the residuals of L_n are precisely those languages belonging to the following family:

$$\mathcal{R} := \{ R_N \mid N \subseteq [0, n] \} \quad \text{where} \quad R_N = (0+1)^* 1 (0+1)^n + \sum_{i \in N} (0+1)^i$$

Proving this claim is sufficient since $|\mathcal{R}| = 2^{n+1}$.

- First we show that every residual of L_n belongs to $\mathcal{R}$. Observe that $L_n = R_\emptyset$ and so $L_n \in \mathcal{R}$. We thus just need to check that $\mathcal{R}$ is closed under taking the residual of words of length one (it follows by induction on word-length that the residual of a word of any length belongs to the family).

  Below we use bold-face for members of the alphabet $\{ \mathbf{0}, \mathbf{1} \}$ to distinguish them from the integers 0 and 1.

$$\begin{aligned} R_N^{\mathbf{0}} &= ((0+1)(0+1)^*)^{\mathbf{0}} + (1(0+1)^n)^{\mathbf{0}} + \sum_{i \in N} ((0+1)^i)^{\mathbf{0}} \\ &= (0+1)^* 1 (0+1)^n + \emptyset + \sum_{i \in \{ j \in N \mid j > 0 \}} (0+1)^{i-1} = R_{\{ j-1 \in N \mid j > 0 \}} \end{aligned} \tag{1}$$

$$\begin{aligned} R_N^{\mathbf{1}} &= ((0+1)(0+1)^*)^{\mathbf{1}} + (1(0+1)^n)^{\mathbf{1}} + \sum_{i \in N} ((0+1)^i)^{\mathbf{1}} \\ &= (0+1)^* 1 (0+1)^n + (0+1)^n + \sum_{i \in \{ j \in N \mid j > 0 \}} (0+1)^{i-1} = R_{\{ n \} \cup \{ j-1 \in N \mid j > 0 \}} \end{aligned} \tag{2}$$

  Thus $R_N^{\mathbf{0}} \in \mathcal{R}$ and $R_N^{\mathbf{1}} \in \mathcal{R}$, as required.

- We now show the converse, that every member of $\mathcal{R}$ is in fact a residual of L_n. In other words, we must show that for each $N \subseteq [0, n]$ there exists a word $w_N \in \{ \mathbf{0}, \mathbf{1} \}^*$ such that $R_N = L_n^{w_N}$.

  We claim that the word $w_\emptyset := \epsilon$ and when $N \neq \emptyset$, $w_N := b_{\min(N)} \cdots b_n$ fits the bill where

$$b_i := \begin{cases} 0 & \text{if } i \notin N \\ 1 & \text{if } i \in N \end{cases}$$

  We argue by induction on the length of w_N (up to a maximum length of $n+1$) that $L_n^{w_N} = R_N$.

  The base case is when $w_N = \epsilon$ and thus $N = \emptyset$. Since $R_\emptyset = L_n = L_n^\epsilon$, the base case holds.

  For the induction step, we assume as induction hypothesis that for a word $w_N = b_n \cdots b_{\min(N)}$ with $\min(N) > 0$ we have $L_n^{w_N} = R_N$. We must show that $L_n^{w_N b} = R_{N'}$ where

$$N' = \begin{cases} \{ j-1 \mid j \in N \} & \text{if } b = 0 \\ \{ n \} \cup \{ j-1 \mid j \in N \} & \text{if } b = 1 \end{cases}$$

  But $L_n^{w_N b} = R_N^b = R_{N'}$ by equations (1) and (2).

□

(iii) A semi-algorithm for membership can be obtained simply by simulating a run of the 2-way DFA on the candidate word. Since the automaton is deterinisitic, if the word is accepted by the automaton, the simulation will terminate.

If the word is rejected by the automaton due to the unique run moving off either end of the word, then the simulation will also terminate.

However if the word is rejected by the automaton because the unique run is infinitely long, then the simulation will not terminate.

Thus to obtain an algorithm for membership, we must simulate the 2-way DFA on the candidate word *and additionally* for each position i in the word maintain a set of states Q_i in which the automaton has been when its head points at i. The algorithm will additionally terminate and announce that the word is rejected by the automaton if the simulation moves the automaton's head to a position i in state q such that already $q \in Q_i$.

Solution 6.3

(i) **Irregular.** The language corresponding to the relation, which we claim is irregular, is by definition the following:

$$L_1 := \{ \, (a_1, b_1) \cdots (a_l, b_l) \mid a_1 \cdots a_l \in \mathit{lsbf}(n) \text{ and } b_1 \cdots b_l \in \mathit{lsbf}(n^2) \text{ for } n \in \mathbb{N} \, \}$$

To show that L_1 is irregular, we show that it has infinitely many residuals. Suppose for contradiction that it only has finitely many residuals.

For each integer $k \geq 1$, let t_k, u_k and v_k be the following words:

$$t_k := \underbrace{(0,0) \cdots (0,0)}_{k\text{-times}} (1,0) \qquad u_k := \underbrace{(0,0) \cdots (0,0)}_{k\text{-times}} (1,0) \underbrace{(0,0) \cdots (0,0)}_{(k-1)\text{-times}} (0,1) \qquad v_k := \underbrace{(0,0) \cdots (0,0)}_{(k-1)\text{-times}} (0,1)$$

Observe that $\pi_1(u_k) \in \mathit{lsbf}(2^k)$ and that $\pi_2(u_k) \in \mathit{lsbf}(2^{2k})$, and so $u_k \in L_1$ for each $k \geq 1$. It must thus be the case that for each $k \geq 1$ we have $v_k \in L_1^{t_k}$.

Since we are assuming that there are only finitely many residuals, there must exist $k_1, k_2 \geq 1$ such that $k_1 \neq k_2$ and $L^{t_{k_1}} = L^{t_{k_2}}$. But that means that $v_{k_2} \in L^{t_{k_1}}$, which is to say that

$$t_{k_1} \, v_{k_2} \;=\; \underbrace{(0,0) \cdots (0,0)}_{k_1\text{-times}} (1,0) \underbrace{(0,0) \cdots (0,0)}_{(k_2-1)\text{-times}} (0,1) \;\in\; L_1$$

This implies that $2^{2k_1} = 2^{k_1+k_2}$ whence $2k_1 = k_1 + k_2$ and $k_1 = k_2$, which is a contradiction. □

(ii) **Irregular.** Now consider the following relation:

$$R_2' := \{ \, (4^k, m) \mid (4^k, m) \in R_2 \text{ for some } k \geq 1 \text{ and there is no } m' \text{ s.t. } (4^k, m') \in R_2 \text{ and } m' > m \, \}$$

Suppose that $(4^k, m) \in R_2'$. Then $(4^k, m) \in R_2$ and so $m \leq \sqrt{4^k}$, that is $m \leq 2^k$. Moreover, there is no $m' > m$ such that $m' \leq 2^k$. Thus in fact $m = 2^k$. So all elements of R_2' must be of the form $(4^k, 2^k)$ for some $k \geq 2$.

Conversely, since $4^k = (2^k)^2$, it follows that all elements of the form $(4^k, 2^k)$ for $k \geq 2$ belong to R_2'. Thus we can characterise R_2' by:

$$R_2' = \{ \, (n^2, n) \mid n = 2^k \text{ for some } k \geq 1 \, \}$$

A very similar argument to that for part (i) shows that R_2' is not regular.

Now suppose for contradiction that R_2 is regular. The relation $<$ is also regular [needs proof, but we leave as an exercise] and so the *join* construction from the lectures tells us that $R_2 \circ <$ is regular. Thus $R_2 \cap \overline{(R_2 \circ <)}$ is regular. But by definition we must have

$$R_2' = \{ \, (4^k, m) \mid k \geq 1 \text{ and } (4^k, m) \in R_2 \cap \overline{(R_2 \circ <)} \, \}$$

Since $\bigcup_{k=1}^{\infty} \mathit{lsbf}(4^k)$ is also regular, we would get that R_2' is regular, a contradiction. □

For the next parts we will first prove that the following relation is regular:

$$R_+ \{ \, (m_1, m_2, n \mid n = m_1 + m_2, \text{ for } n, m_1, m_2 \in \mathbb{N} \, \}$$

Note that this is a *ternary relation* and so its elements will be encoded by words over the alphabet $\{ \, 0, 1 \, \}^3$ (instead of $\{ \, 0, 1 \, \}^2$, as is used for binary relations).

We claim that the DFA $A_+ \langle Q_+, \{ 0,1 \}, \hat{0}, \delta_+, F_+ \rangle$ recognises the language encoding R_+, where $Q_+ = \{ \hat{0}, \hat{1}, X \}$, $F_+ = \{ \hat{0} \}$, and δ is given by

$$\delta_+(\hat{c}, (b_1, b_2, b)) := \hat{c'} \quad \text{where} \quad b = \begin{cases} 0 & \text{if an even number of } c, b_1, b_2 \text{ are } 0 \\ 1 & \text{if an odd number of } c, b_1, b_2 \text{ are } 0 \end{cases} \quad c' = \begin{cases} 0 & \text{if fewer than two of } c, b_1, b_2 \text{ are } 1 \\ 1 & \text{if at least two of } c, b_1, b_2 \text{ are } 1 \end{cases}$$

and $\delta_+(\hat{c}, (b_1, b_2, b)) = X$ otherwise. Intuitively A_+ implements a half-adder. It represents the 'carry bit' in its state and sums this together with the input bits b_1 and b_2 to get the output bit b and the state of the new carry bit. (The sink state X is adopted when it reads an incorrect output bit).

We can formulate this intuition with the following assertion: After reading a word of the form $(a_1, b_1, c_1) \cdots (a_k, b_k, c_k)$ (for $k \geq 0$) such that $a_1 \cdots a_k \in \mathit{lsbf}(m_1)$ and $b_1 \cdots b_k \in \mathit{lsbf}(m_2)$, and $c_1 \cdots c_k \in \mathit{lsbf}(n)$, it will be the case that:

- if $n = (m_1 + m_2)$, then the automaton will be in state $\hat{0}$
- if $n + 2^k = (m_1 + m_2)$, then the automaton will be in state $\hat{1}$
- otherwise the automaton will be in state X.

This assertion can be proved by induction on the length k of this word. The base case in when $k = 0$, and here the assertion holds since $\epsilon \in \mathit{lsbf}(0)$ and after 'reading' the empty word the automaton remains in its initial state, which is $\hat{0}$.

For the induction step let us suppose that the automaton has read a word $w = (a_1, b_1, c_1) \cdots (a_k, b_k, c_k)(a_{k+1}, b_{k+1}, c_{k+1})$ of length $k+1$. Let $A_k \in \mathit{lsbf}(a_1 \cdots a_k)$, $A_{k+1} \in \mathit{lsbf}(a_1 \cdots a_k, a_{k+1})$ $B_k \in \mathit{lsbf}(b_1 \cdots b_k)$, $B_{k+1} \in \mathit{lsbf}(b_1 \cdots b_k, b_{k+1})$ $C_k \in \mathit{lsbf}(c_1 \cdots c_k)$, $C_{k+1} \in \mathit{lsbf}(c_1 \cdots c_k, c_{k+1})$.

Observe that it is always the case that $A_{k+1} + B_{k+1} = A_k + B_k + a_{k+1}.2^k + b_{k+1}.2^k$ and that $C_{k+1} = C_k + c_{k+1}.2^k$. We consider each of several cases in turn.

- Suppose first that $C_{k+1} = A_{k+1} + B_{k+1}$. Then $C_k + c_{k+1}.2^k = A_k + B_k + a_{k+1}2^k + b_{k+1}2^k$. It must then be the case that one of the following holds:
  - $C_k = A_k + B_k$. Then by the induction hypothesis the automaton will be in state $\hat{0}$ prior to reading the symbol $(a_{k+1}, b_{k+1}, c_{k+1})$. Also it must be the case that $c_{k+1}.2^k = a_{k+1}2^k + b_{k+1}2^k$.

    Suppose first that $c_{k+1} = 0$. Then it must be the case that $a_{k+1} = b_{k+1} = 0$. Then it must be the case that $\delta_+(\hat{0}, (a_{k+1}, b_{k+1}, c_{k+1})) = \hat{0}$.

    Suppose now that $c_{k+1} = 1$. Then it must be the case that $\{ a_{k+1}, b_{k+1} \} = \{ 0, 1 \}$ and that $\delta_+(\hat{0}, (a_{k+1}, b_{k+1}, c_{k+1})) = \hat{0}$.

    So in these cases we end up in state $\hat{0}$ and we also satisfy the requirement that $C_{k+1} + 0.2^{k+1} = A_{k+1} + B_{k+1}$.
  - $C_k + 2^k = A_{k+1} + B_{k+1}$. By the induction hypothesis the automaton will be in state $\hat{1}$ prior to reading the symbol $(a_{k+1}, b_{k+1}, c_{k+1})$. Also it must be the case that $c_{k+1}.2^k = 2^k + a_{k+1}.2^k + b_{k+1}.2^k$. Thus it must be the case that $c_{k+1} = 1$ and $a_{k+1} = b_{k+1} = 0$.

    Thus again by the definition of δ_+ we end up in state $\hat{0}$ with the required $C_{k+1} + 0.2^{k+1} = A_{k+1} + B_{k+1}$.
- Suppose now that $C_{k+1} + 2^{k+1} = A_{k+1} + B_{k+1}$. We can perform a similar case analysis to the above to get that we end up in state $\hat{1}$ (with the required $C_{k+1} + 2^{k+1} = A_{k+1} + B_{k+1}$). This will use the fact that $2^k + 2^k = 2.2^k = 2^{k+1}$.
- The final case is when neither $C_{k+1} = A_{k+1} + B_{k+1}$ nor $C_{k+1} + 2^{k+1} = A_{k+1} + B_{k+1}$. We can again perform a case analysis to check that we must end up in state X. (This time we must check three possibilities for the induction hypothesis. $C_k = A_k + B_k$, $C_k + 2^k = A_k + B_k$ and 'other', giving us states $\hat{0}$, $\hat{1}$ and X respectively immediately after reading the kth element.

Since $\hat{0}$ is the only final state, it follows immediately that $\mathcal{L}(A_+) \subseteq \mathcal{L}(R_+)$. We can also see that $\overline{\mathcal{L}(A_+)} \subseteq \overline{\mathcal{L}(R_+)}$ since for $w \in \overline{\mathcal{L}(A_+)}$ either a run of A_+ in w will end in $\hat{1}$, in which case the assertion tells us that w represents a triple of the form $(m_1, m_2, m_1 + m_2 + 1)$, or else it will end in state X, which tells us in particular that w does not represent a triple of the form $(m_1, m_2, m_1 + m_2)$.

Thus $\mathcal{L}(A_+) = \mathcal{L}(R_+)$. □

For technical reasons it will be helpful to consider a variant of R_+ which we call $R_+^=$. This differs in two ways: (i) it makes a copy of the first number being summed, and (ii) although $R_+^=$ can naturally be seen as a quarternary relation, we in fact view it as a binary relation consisting of pairs of pairs (this will allow us to use *Post* below):

$$R_+^= := \{ ((m_1, m_2), (m_1, m_1 + m_2)) \mid m_1, m_2 \in \mathbb{N} \}$$

The regularity of $R_+^=$ follows from the regularity of R_+.

(iii) **Regular** We prove something a bit more general: for every $\lambda \in \mathbb{N}$ it is the case that the relation

$$R_\lambda := \{ (n, \lambda.n) \mid n \in \mathbb{N} \}$$

is regular. (The relation in the question is the special case when $\lambda = 5$).

We argue by induction on $\lambda \in \mathbb{N}$. The base case is when $\lambda = 0$, which is trivial, since $R_0 = \{ (n, 0) \mid n \in \mathbb{N} \} = (\{ 0, 1 \} \times \{ 0 \})^*$.

For the induction step, suppose that we have already shown that R_λ is regular. We can view R_λ as a regular set of pairs of numbers and we have

$$R_{\lambda+1} = \{ (n, n + \lambda.n) \mid n \in \mathbb{N} \} = \{ p \mid ((n, \lambda.n), p) \in R_+^= \} = \{ p \mid (p_0, p) \in R_+^= \text{ for some } p_0 \in R_\lambda \} = Post_{R_+^=}(R_\lambda)$$

Since $R_+^=$ and R_λ are both regular, the lectures tell us that $Post_{R_+^=}(R_\lambda)$ and thus $R_{\lambda+1}$ is also regular. $\square$

(iv) **Regular** The set of pairs $L_7 = \{ (n, 7) \mid n \in \mathbb{N} \}$ is regular since it is represented by the language given by the following regular expression

$$\left(\sum_{b_1, b_2, b_3 \in \{ 0,1 \}} (b_1, 1)(b_2, 1)(b_3, 1) \right) (\{ 0, 1 \} \times \{ 0 \})^*$$

The required language is thus given by $Post_{R_+^=}(L_7)$. $\square$

(v) **Regular** The required relation is given by taking the join $R_5 \circ L_7$ of the relations from (iii) and (iv). We saw in lectures that taking the join of regular relations produces a regular relation. $\square$

(vi) **Regular** Let us consider the following variant of R_+ (which must also be regular):

$$\widehat{R_+} := \{ (n_1, m_1), (n_2, m_2), ((n_1, n_2), m_1 + m_2) \mid n_1, n_2, m_1, m_2 \in \mathbb{N} \}$$

Note that n_1 and n_2 are only treated trivially by $\widehat{R_+}$ (they are copied).

We then get

$$P := \{ ((n, m), 2n + 3m) \mid n, m \in \mathbb{N} \} = Post_{Post_{\widehat{R_+}}(R_2)}(R_3)$$

(using R_2 and R_3 from part (iii)). Thus P is also regular.

The language $K := 0^*10^*$ is regular and encodes the set $\{ 2^k \mid k \in \mathbb{N} \}$. The required relation is equal to

$$Pre_P(K)$$

and is thus regular (using the theorem concerning the preservation of regularity by *Pre* from the lectures). $\square$