Grading key

Grade	Points
1.0	≥ 59
1.3	≥ 56
1.7	≥ 53
2.0	≥ 50
2.3	≥ 47
2.7	≥ 44
3.0	≥ 41
3.3	≥ 38
3.7	≥ 35
4.0	≥ 32

Problem 1

0P if the answer is wrong or no justification is given

Problem 1a

- **2P** for a correct counterexample
- **2P** for a counterexample which is correct for the alphabet $\{a\}$
- **1.5P** if the DFA would have a trap state which makes the counterexample wrong, but this can be fixed by adding a trap state to the NFA
- **1P** if the counterexample can be fixed by making a state initial
- **-0.5P** if the *NFAtoDFA* algorithm is executed incorrectly

Problem 1b

- $+$**1P** for creating a new final state q_f without outgoing transitions
- $+$**1P** for adding ε-transitions and $+$**1P** for removing them; alternatively, $+$**0.5P** for copying transitions to previously final states to q_f and $+$**1.5P** for adding transitions from q_f to successors of initial states
- **1.5P** for the construction which just adds (f, a, q) for every transition (q_0, a, q) for all $f \in F$ and $q_0 \in Q_0$. This construction does not work as the NBA is not forced to use the extra transitions. For example, applying it to the 2-state NFA recognizing $L(ab^*)$ gives an NBA which erroneously accepts ab^ω.

Problem 1d

- **1P** for converting the NBA into an NRA with Rabin pair $\langle F, Q \setminus F \rangle$

Problem 1f

- **1.5P** for $\{p\}\{r\}$ (this is not a computation in $(2^{\mathsf{AP}})^\omega$)
- **1.5P** for pr^ω (p and r are not part of the alphabet)

Problem 2

Problem 2a

- **1.5P** if the NFA recognizes $L_n \setminus \{\varepsilon\}$

Problem 2b

- **+1.5P** for the statement that the number of states/residuals is equal to the number of subsets of Σ_n
- **+2P** for the statement that given $u, v \in \Sigma_n^*$ such that the set of letters in u is not equal to the set of letters in v, $L_n^u \neq L_n^v$ holds
- **+1.5P** for the proof of that statement, e.g. by giving a word in $L_n^u \setminus L_n^v$

Problem 3

Problem 3a

- **0P** if no attempt at a product construction is apparent
- **-0.5P** if the set of initial states is wrong
- **-0.5P** if the set of final states is wrong
- **-0.5P** for each missing/incorrect transition; this includes missing transitions from undiscovered states

Problem 3b

- **1P** for $\{\{q_1\} \times Q_2, \{q_3\} \times Q_2\}$ or $\{\{q_1\} \times \{r_1, r_3\}, \{q_3\} \times \{r_1, r_3\}\}$
- **1P** for $\{\{q_1\}, \{q_3\}, \{r_1, r_3\}\}$
- **0P** for $\{\{q_1, q_3\} \times \{r_1, r_3\}\}$ or $\{\{q_1 r_1\}, \{q_1 r_3\}, \{q_3 r_1\}, \{q_3 r_3\}\}$

Problem 3c

- **+1P** for drawing a one-state NGA accepting $\emptyset$ or stating that $L_\omega(B_1) \cap L_\omega(B_2) = \emptyset$
- **+2P** for the justification that $L_\omega(B) = \emptyset$ either by arguing that states containing q_1 (the top row) cannot be visited infinitely often or by arguing that words must contain aa infinitely often to be accepted by B_1, but such words are not accepted by B_2
  - **+1P** instead if this justification is made significantly easier by a mistake in a previous subproblem, e.g. if you have $\{\{q_1\} \times \{r_1, r_3\}, \{q_3\} \times \{r_1, r_3\}\}$ as the acceptance condition in b)

Problem 4

Problem 4a

- **-0.5P** for each unnecessary transition
- **1P** if the NFA recognizes $\Sigma^* p \Sigma^*$

Problem 4b

- **-0.5P** for each missing/incorrect transition
- **-0.5P** if the NFA in a) recognizes $\Sigma^* p \Sigma^*$ and the powerset construction is executed incorrectly after reaching the final state of the NFA (the DFA should then have six final states)

Problem 4c

- the solution is graded by comparing it to the correct solution; mistakes resulting from a wrong DFA in b) are treated the same way as other mistakes
- **-1P** for each missing *miss* transition, or *miss* transition with the incorrect target state
- **-0.5P** for writing N instead of R on the first *miss* transition
- **-0.5P** for not writing N or R at all

Problem 5

Problem 5a

- **+1P** for giving an infinite set $\{w_n \mid n \in \mathbb{N}\}$ of words such that $R_1^{w_i} \neq R_1^{w_j}$ for $i \neq j$
- **+3P** for proving that $R_1^{w_i} \neq R_1^{w_j}$ for $i \neq j$. A working strategy is to give a word in $R_1^{w_i} \Delta R_1^{w_j}$, or to explicitly compute $R_1^{w_i}$ and $R_1^{w_j}$.

Problem 5b

- **-0P** if the transducer does not accept ε
- **-0.5P** if the transducer does not accept $\left[\begin{smallmatrix} a \\ a \end{smallmatrix}\right]$ or $\left[\begin{smallmatrix} b \\ b \end{smallmatrix}\right]$
- **-0.5P** for each missing/incorrect transition

Problem 6

- **+1P** if the answer and justification is correct

Problem 7

Problem 7a

- **-1P** for each missing residual. Exception: **-0.5P** if r is missing

Problem 7c

- **+1P** for 1. and 2., **+2P** for 3.

Problem 7d

- **3P** if the DBA accepts words of the form $(ac)^n c^\omega$, but is otherwise correct
- **1P** if the DBA rejects b^ω and c^ω, but is otherwise correct

Problem 7f

- **1.5P** for $\{p\}^*\{p,q\}(\{p\} + \{p,q\})^\omega$
- **0.5P** for ω-regular expressions equivalent to $p \ \mathbf{U} \ q$, e.g. $\{p\}^*(\{q\} + \{p,q\})\Sigma^\omega$