Solutions

Marking: **M1** a correct method, **A1** the correct answer, **B1** an independent mark.

Each answer shows the steps, then the **result**.

2.1

q0, q1, q2, q0, q1, q2

ends in q2: accepted

all strings that end in `ab`

2.2

`(a|b)*aa(a|b)*`

`(a|b)*b`

2.3

from 0: 0→0, 1→1

from 1: 0→2, 1→0

from 2: 0→1, 1→2; accept in state 0

2.4

suppose a DFA with p states did: on a^p it must visit some state twice

so the loop of as can be repeated: $a^{p+k}b^p$ would also be accepted, which is wrong

3.1

(a) {A}: 0→{A}, 1→{A,B}; {A,B}: 0→{A,C}, 1→{A,B,C}; {A,C}: 0→{A}, 1→{A,B}; {A,B,C}: 0→{A,C}, 1→{A,B,C}: 4 states

(b) the sets that contain C: {A,C} and {A,B,C}

(c) 2^k: the DFA must remember the last k symbols

(d) simulate the NFA directly, keeping the set of current states, or build DFA states only when they are needed

3.2

(a) 0: a→1, b→0; 1: a→1, b→2; 2: a→1, b→3; 3 (accepting): a→1, b→0

(b) 0, 0, 1, 2, 3: accepted

(c) the strings ε, `a`, `ab`, `abb` need different futures (e.g. adding `bb`, `b` or nothing), so each needs its own state

3.3

(a) `[A-Z]{2}[0-9]{6}`, anchored to the whole input: `^…$`

(b) one state for each of 0–8 characters read (9 states), plus one dead state: **10**

(c) $26^2 \times 10^6 = \mathbf{6.76 \times 10^8}$

exercises.lol exercises.com.cn

(d) brackets can nest to any depth, like a^nb^n, so the language is not regular; a DFA cannot count unbounded depth; use a short loop with a counter (add 1 for `(`, subtract 1 for `)`, never below 0, end at 0)