Solutions

2.1

- q_2 ✓.

2.2

- It ends in state q_1 – the final `a` leaves the pattern incomplete ✓.

2.3

- It forbids unbounded counting: every long string has a pumpable loop ✓.

3.1

- `babbab`: q0→q0 (b), q0→q1 (a), q1→q2 (b), q2→q0 (b), q0→q1 (a), q1→q2 (b) – accepted ✓.
- `babba`: q0, q0, q1, q2, q1, q1 – rejected ✓.
- Swapping the last two letters flips it: `babbab` keeps ...ab; `babba...` variants end ba or aa – only strings ending ab accept, so the final pair decides ✓.
- A complete table over $\{a, b, c\}$ needs a dead state q_d: q_d loops on every letter, entered on c from anywhere ✓.

3.2

- Choose $a^p b^p$ ($n = p$ forced by the lemma's threshold) ✓.
- $|xy| \le p$ keeps y inside the a-block: $y = a^k$, $k \ge 1$ ✓.
- Pump once: $xy^2z = a^{p+k}b^p$ – more a's than b's, not in L ✓.
- Ceiling for lexers: nested brackets need counting, so no regex (DFA) lexer can match them – the parser above must take over ✓.