

UM SISTEMA PARA AUXILIAR NA APRENDIZAGEM DA DISCIPLINA LINGUAGENS FORMAIS E AUTÔMATOS



Rafael Cardoso da Silva, Daniel Martin Morgato
Centro de Matemática, Computação e Cognição, Universidade Federal do ABC
Av. dos Estados, 5001, Santo André, SP
rafael.cardoso@aluno.ufabc.edu.br, daniel.martin@ufabc.edu.br

Resumo: Resolver exercícios é fundamental para um aluno fixar os conceitos apresentados em aula. Por outro lado, ter seus exercícios corrigidos também é muito importante, para que ele possa avaliar o seu aprendizado. Na UFABC, a disciplina de Linguagens Formais e Autômatos contempla vários exercícios que admitem infinitas respostas, o que torna a correção deles praticamente impossível, principalmente quando as turmas são grandes. O objetivo deste projeto foi a criação e implementação de um sistema para aplicação e correção automática de exercícios envolvendo autômatos finitos determinísticos. Através do estudo de métodos e algoritmos presentes na literatura, foi possível implementar o teste de equivalência entre o autômato-resposta do aluno e o autômato-gabarito previamente armazenado no banco de dados. Ao final do projeto, o sistema foi usado em caráter experimental numa turma da UFABC da disciplina de Linguagens Formais e Autômatos, afim de testar a sua qualidade. E ao final da disciplina, a nota que os alunos obtiverem ao revolver os exercícios do sistema ajudarão a compor o conceito final de cada um na disciplina.

Palavras-chave: autômato finito, equivalência de autômatos, minimização de autômato, programação para web.

INTRODUÇÃO

Na disciplina MC3106 – Linguagens Formais e Autômatos – vários exercícios da parte inicial da disciplina pedem como resposta um autômato finito determinístico. Tais exercícios são particularmente trabalhosos de se corrigir porque cada um deles admite infinitas respostas corretas e, por esse motivo, sua correção se torna exaustiva para o monitor ou docente responsável.

Com o intuito de agilizar essa tarefa, este projeto propõe o desenvolvimento de um sistema web que permita ao aluno inserir seu autômato-resposta por meio de uma interface intuitiva e fácil de ser utilizada e obter um *feedback* quase que imediatamente, um tempo drasticamente reduzido se comparado à correção manual.

Um *autômato finito determinístico* (também chamado de AFD ou máquina de estados) consiste de um conjunto de estados não vazio Q , um alfabeto Σ , um estado inicial $s \in Q$, um conjunto de estados de aceitação $F \subseteq Q$ e uma função de transição $\delta: Q \times \Sigma \rightarrow Q$. É comum estender a função de transição para uma função $\hat{\delta}: Q \times \Sigma^* \rightarrow Q$ definida indutivamente por

- (i) $\hat{\delta}(q, \varepsilon) = q$ para todo estado $q \in Q$;
- (ii) $\hat{\delta}(q, \sigma) = \delta(q, \sigma)$ para todo estado $q \in Q$ e símbolo $\sigma \in \Sigma$;
- (iii) $\hat{\delta}(q, \sigma_1 \dots \sigma_k) = \delta(\hat{\delta}(q, \sigma_1 \dots \sigma_{k-1}), \sigma_k)$ para todo estado $q \in Q$ e símbolos $\sigma_1, \dots, \sigma_k \in \Sigma$.

Abaixo vemos dois exemplos de AFDs que reconhecer a seguinte linguagem:

$$L() = \{w \in \Sigma^* : w \text{ tem número par de símbolos } a\}$$

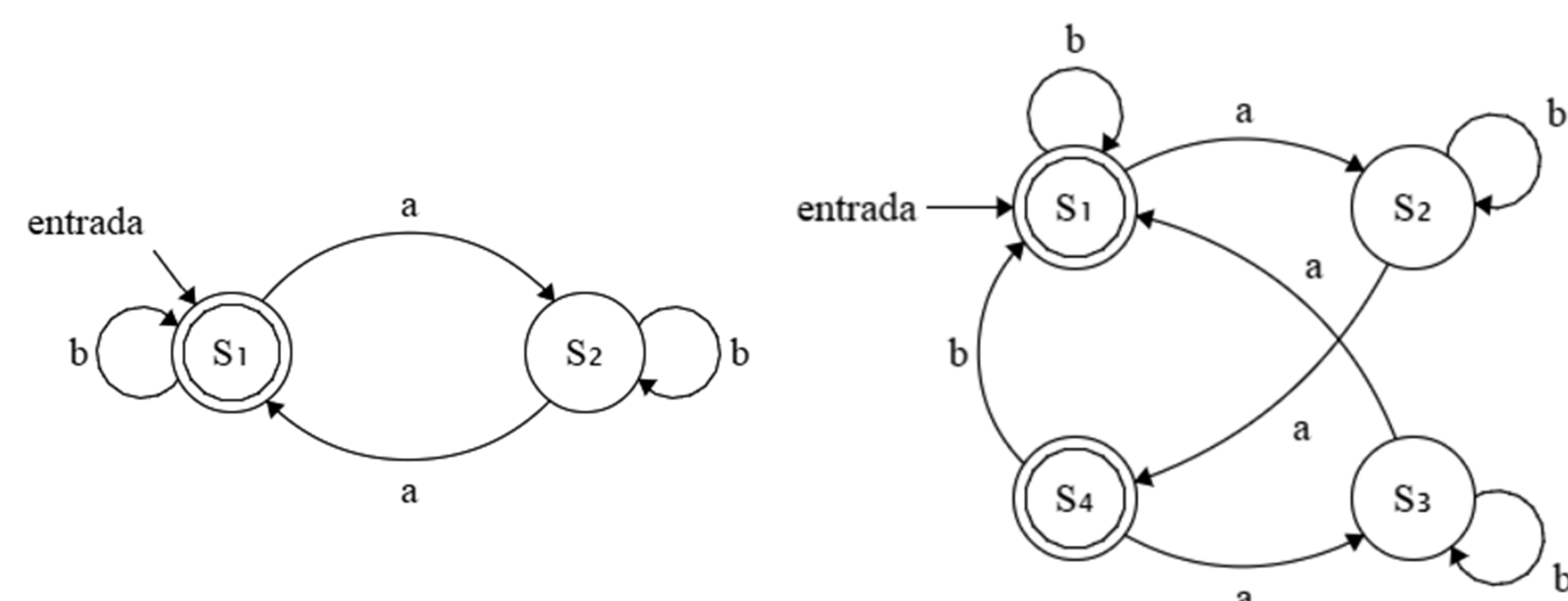


Figure 1: Autômato M_1 a esquerda e Autômato M_2 a direita.

Apesar de M_2 ser visivelmente maior e mais complexo que M_1 , o autômato M_2 reconhece a mesma linguagem que o autômato M_1 da Figura 1. É possível demonstrar que, para cada linguagem regular, existem infinitos autômatos que a reconhecem. Como decidir então se o autômato-resposta dado por um aluno reconhece a linguagem pedida? No caso do sistema que iremos desenvolver isso se reduz ao seguinte problema:

Problema: Decidir se dois autômatos finitos determinísticos reconhecem a mesma linguagem.

OBJETIVOS E METODOLOGIA

Estudar e implementar o algoritmo de Moore (MOORE, 1956) para minimização e equivalência de AFDs.

Estudar e implementar o algoritmo de Hopcroft e Karp (HOPCROFT; KARP, 1971) para testar a equivalência de AFDs;

Desenvolvimento do sistema de apoio a aprendizagem da disciplina Linguagens Formais e Autômatos

Reuniões semanais com o orientador, afim de acompanhar o progresso do desenvolvimento do sistema e esclarecer eventuais dúvidas.

RESULTADOS

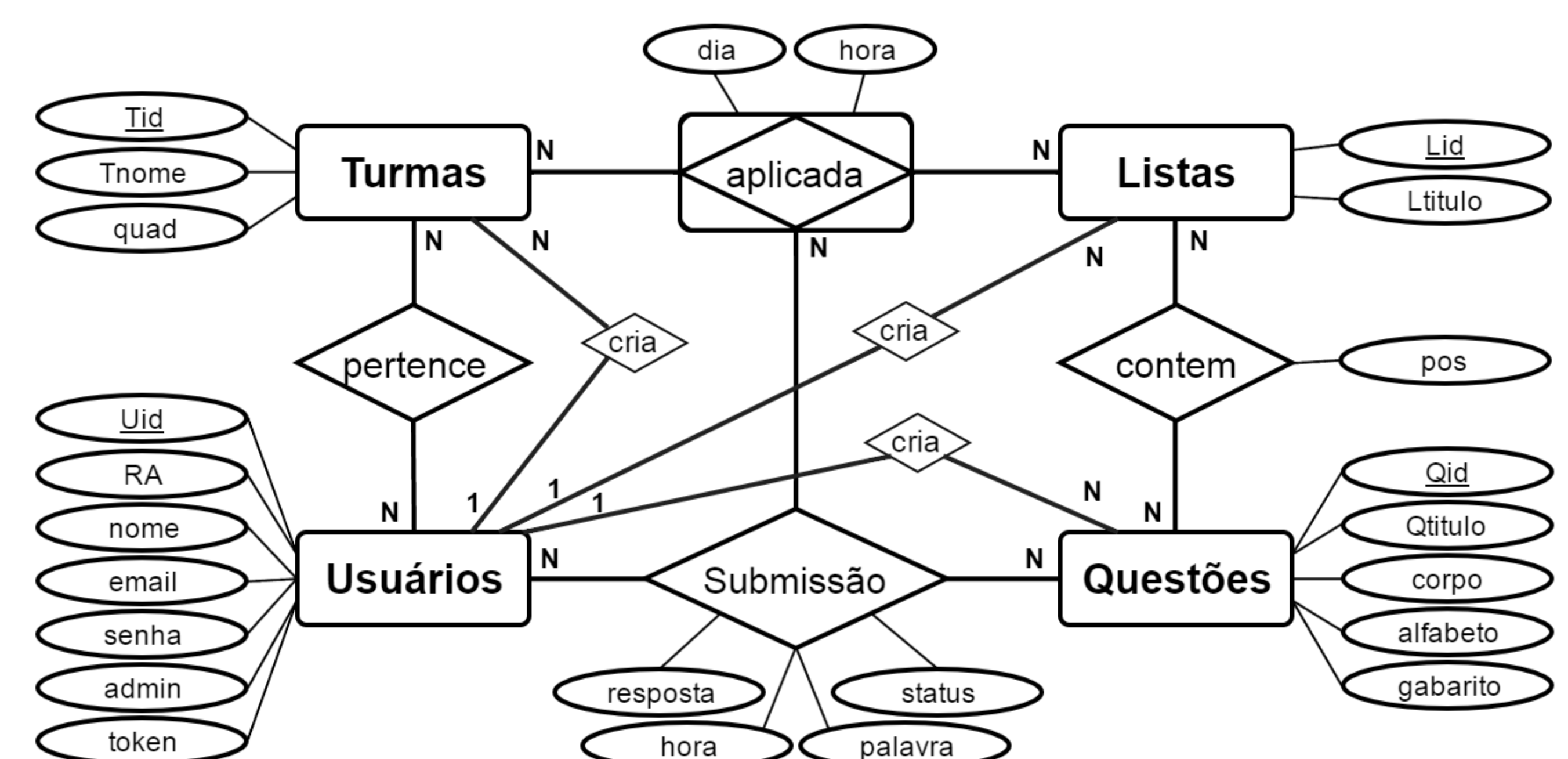


Figure 2: Diagrama de Entidade e Relacionamento do DFAjudge

Tipos de Feedback

Teste com a Turma de LFA 2016q2

DISCUSSÃO

CONCLUSÃO

REFERÊNCIAS

HOPCROFT, J. E.; MOTWANI, R.; ULLMAN, J. D. *Automata theory, languages, and computation*. International Edition, v. 24, 2006.

MOORE, Edward F. *Gedanken-experiments on sequential machines*. Automata studies, v. 34, p. 129-153, 1956.

HOPCROFT, John E.; KARP, Richard M. *A Linear Algorithm for Testing Equivalence of Finite Automata*. Technical report of Cornell University, p. 71–114, 1971.

AGRADECIMENTOS

Este trabalho foi financiado pelo Programa de Iniciação Científica da UFABC.