

## Linguagens Formais e Autômatos — Trabalho

O trabalho, individual, consiste em um relatório e na codificação de dois problemas.

O ambiente de programação é, preferencialmente, o Scilab. Se não quiser utilizar o Scilab, pode usar, nesta ordem de preferência, o Matlab, o Python, ou qualquer outro ambiente que considere adequado.

A entrega do trabalho deve ser realizada no ambiente Moodle, no *link* de *upload* a isso destinado. O trabalho deve conter, em um único arquivo zipado, os seguintes arquivos:

- README — arquivo txt dizendo o que está no arquivo zipado, o ambiente computacional utilizado, e como fazer para rodar os programas;
- um arquivo PDF com o relatório
- arquivos com os códigos-fonte dos programas
- arquivos com os dados de entrada e de saída utilizados na ilustração do funcionamento dos programas
- além de qualquer outro material que ache necessário incluir.

O relatório deve ser feito seguindo o formato disponível na aba Moodle do trabalho, no endereço <https://ead.iprj.uerj.br/moodle/mod/url/view.php?id=10784>, e você deve ilustrar o funcionamento dos programas.

Os exercícios computacionais, retirados do livro-texto, são apresentados a seguir.

**Programa 1** Escreva um simulador de máquina de estado finito. Isto é, dado

- um número inteiro positivo  $n$ ,  $n \leq 50$ , representando o número de estados de uma máquina de estados finitos com alfabeto de entrada = alfabeto de saída =  $\{0, 1\}$ ,
- uma matriz  $n \times 3$  que representa a descrição da tabela de estado de tal máquina,

seu programa deve solicitar *strings* de entrada e escrever as *strings* de saída correspondentes pelo tempo que o usuário desejar.

**Programa 2** Escreva um simulador de máquina de Turing. Ou seja, dado um conjunto de quíntuplas que descreve uma máquina de Turing, seu programa deve solicitar o conteúdo inicial da fita e escrever uma sequência de configurações de fita sucessivas.

Suponha que haja no máximo 100 quíntuplas, que o número de células usadas na fita seja no máximo 70, e permita que o usuário defina um número máximo de etapas caso o cálculo não pare antes disso.