Prova substituiva de Teoria da Computação — Campus de Sorocaba da UFSCar
Prof. José de Oliveira Guimarães, 7 de fevereiro de 2013

As questões deverão ser respondidas nas folhas de respostas e apenas estas deverão ser entregues (não entregue a folha de questões). Utilize quantas folhas forem necessárias. Não é necessário entregar as questões em ordem. Utilize lápis ou caneta, mas faça letra legível. Na correção, símbolos ou palavras ilegíveis não serão considerados. Justifique todas as respostas.

Se você estudou com algum colega para esta prova, não se sente ao lado dele pois é possível que acidentalmente vocês produzam respostas semelhantes para alguma questão. Provas de alunos que fizeram a prova próximos uns dos outros com respostas semelhantes caracterizam cópia de questões. Lembro-os de que todos os envolvidos na utilização de métodos ilegais na realização desta prova receberão zero de nota final da disciplina (e não apenas nesta prova).

Coloque o seu nome na folha de resposta, o mais acima possível na folha, seguido do número da sua coluna de carteiras. A primeira carteira é a mais perto da porta e a última a mais perto das janelas. Não precisa colocar o RA.

Todas as questões valem 1,5 ponto. Você deve fazer quatro questões da primeira parte da disciplina e quatro da segunda parte.

Primeira parte da disciplina

1. Faça um AFD que reconheça a linguagem $L = \{a^n b a^k : n \geqslant 0, k \geqslant 1\}$.

2. Faça uma expressão regular R tal que $L(R) = \{0^n 1^m 0^p : n \geqslant 0, m \geqslant 1, p \geqslant 2\}$.

3. Explique porque a linguagem $L = \{0^n 1 0^n 1^{2n} : n \geqslant 1\}$ não pode ser reconhecida por um AF. A explicação pode ser informal, sem citação de nenhum teorema.

4. Faça uma gramática G tal que $L(G) = \{ba^n b : n \geqslant 0\}$.

5. Seja *Reg* o conjunto de todas as linguagens regulares, *LLC* o conjunto das linguagens livres de contexto e *EF* o conjunto das linguagens com estrutura de frase. Faça uma tabela que relacione, para cada um destes conjuntos, o tipo de autômato (ou MT) e a gramática. Como exemplo, na linha do conjunto *Reg* pode aparecer "AFP" e "Gramáticas Limitadas de Contexto" pois uma linguagem de *Reg* pode ser reconhecida por um autômato AFP (e este é o mais simples autômato que pode reconhecer linguagens de *Reg*) e uma gramática da classe "Gramáticas Limitadas de Contexto" é a gramática mais simples que pode produzir uma linguagem de *Reg*.

6. Converta o seguinte AFND para AFD utilizando o algoritmo de conversão apresentado no

livro texto do M. Sipser utilizado em aula. Isto é, você não pode descobrir a linguagem que o AFND gera e, usando a sua intuição, fazer o AFD.

$$(q_0, 1, \{q_1\}), (q_0, 0, \{q_0, q_1\}), (q_1, a, \{q_1\}), (q_1, \epsilon, \{q_0\})$$

Segunda parte da disciplina

7. Cite a tese de Church-Turing e explique-a.

8. Seja R o conjunto das linguagens recursivas (computáveis), RE o conjunto das linguagens recursivamente enumeráveis (computacionalmente enumeráveis), L_p uma linguagen contendo os números que são potência de 2, P o conjunto das linguagens decidíveis em tempo polinomial por uma MTD, NP o conjunto das linguagens decidíveis em tempo polinomial por uma MTND. Cite todos os relacionamentos entre estes conjuntos com relação a $\subset$ e $\in$.

9. Prove que, se L é decidível, L^c é decidível também.

10. Na simulação de uma MTD de k fitas por uma MTD de uma fita, como podem ser representadas as k fitas em uma única? Cite apenas a representação, não é necessário explicar como a simulação funciona.

11. A linguagem $L = \{0^n 1 : n \geqslant 0\}$ é recursivamente enumerável? Justifique.

12. A MT P toma dois inteiros como parâmetros. O primeiro deles é uma codificação de uma MT M e o segundo uma entrada x para esta MT (assim P interpreta as suas entradas). P retorna 1 se $M(x) \downarrow$ e 0 se $M(x) \uparrow$. Usando P, construimos a MT Q que toma dois inteiros m e x como parâmetros e:

(a) simula a execução de P com os parâmetros m e x;

(b) se P retorna 1, Q entra em um laço infinito, nunca terminando a sua execução;

(c) se P retorna 0, Q termina a sua execução.

O que acontece se chamamos Q passando $\langle Q \rangle$ e $\langle Q \rangle$ como parâmetros? Explique a contradição e o que podemos deduzir dela.