Teoria da Computação

Prova 2 - 05/07/2010 - Prof. Marcus Ramos - UNIVASF

1. (1 ponto) Conceitue:
   a. Problema solucionável;

      Problema que é representado por uma linguagem recursiva.

   b. Problema não-solucionável;

      Problema que é representado por uma linguagem não-recursiva.

   c. Problema parcialmente solucionável;

      Problema que é representado por uma linguagem recursivamente enumerável.

   d. Problema totalmente insolúvel.

      Problema que é representado por uma linguagem não-recursivamente enumerável.

2. (1 ponto) Se L é uma linguagem recursiva (respectivamente recursivamente enumerável e não-recursivamente enumerável), qual é o tipo mais restrito de ~L (o complemento de L)? Justifique as suas respostas.

- **Se L é recursiva, o seu complemento é recursivo também.**
- **Se L é recursivamente enumerável, seu complemento pode ser recursivo (se L for recursiva) ou não-recursivamente enumerável (se L for não-recursiva).**
- **Se L é não-recursivamente enumerável, seu complemento poderá ser recursivamente enumerável não-recursiva ou não-recursivamente enumerável.**

3. (1 ponto) Considere uma Máquina de Turing Universal. Descreva:
   a. A sua arquitetura;

      Possui quatro fitas de entrada independentes. A descrição da máquina a ser simulada e a sua entrada, são gravadas na primeira fita. A segunda fita é usada para representar a fita de entrada da máquina sendo simulada, a terceira fita representa o estado corrente da máquina sendo simulada e a quarta fita é usada como rascunho.

   b. A sua operação;

      A Máquina Universal copia a cadeia de entrada da máquina sendo simulada para a segunda fita. Em seguida, são analisados os símbolos dessa fita e executados os movimentos conforme as especificações da máquina que está sendo simulada e que estão armazenadas na primeira fita.

   c. A linguagem por ela aceita.

A linguagem aceita pela MTU é a Linguagem Universal, formada por todas as cadeias <M,w> que são obtidas pela justaposição de codificações de Máquinas de Turing com cadeias de entrada que são aceitas por essas Máquinas de Turing.

4. (0,5 ponto) O que é uma redução? Exemplifique.

   Uma redução f de uma linguagem A para uma linguagem B é uma função (total, porém não necessariamente sobrejetora) que mapeia cadeias de A em cadeias de B, ou seja, $w \in A \Leftrightarrow f(w) \in B$.

5. (1 ponto) De que forma uma redução deve ser usada para:
   a. Provar que um certo problema é solucionável?

      Deve-se reduzir esse problema para um segundo problema sabidamente solucionável.

   b. Provar que um certo problema é não-solucionável?

      Deve-se reduzir um segundo problema sabidamente não-solucionável para esse problema.

6. (1 ponto) Prove que o problema {P | P é decidível} é indecidível.

   De acordo com o Teorema de Rice, toda propriedade não-trivial das linguagens recursivamente enumerável é indecidível. Assim, basta provar que "ser decidível" é uma propriedade não-trivial das mesmas:

   - **Todas as linguagens livres de contexto são decidíveis (portanto, o conjunto das linguagens que apresentam a propriedade é não-vazio);**
   - **A linguagem Lu é recursivamente enumerável e não-recursiva (portanto, o conjunto das linguagens que não apresentam a propriedade é não-vazio).**

7. (1 ponto) Justifique o fato de o problema do pertencimento ser solucionável para autômatos linearmente limitados e não-solucionável para Máquinas de Turing.

   No caso dos autômatos linearmente limitados (Máquinas de Turing com fita limitada), a quantidade de configurações que eles podem assumir durante a análise de uma cadeia é sempre finita, o que não acontece com a Máquina de Turing genérica. Dessa forma, basta verificar se uma configuração se repete para identificar um loop e rejeitar a cadeia de entrada. Na Máquina de Turing genérica essa verificação não pode ser feita no caso geral.

8. (0,5 ponto) Descreva dois problemas que se demonstra serem não-solucionáveis a partir de reduções efetuadas com base em PCP.

   - **Determinar se uma GLC é ambígua;**
   - **Determinar se as linguagens geradas por duas GLCs são disjuntas;**
   - **Determinar se as linguagens geradas por duas GLCs são idênticas;**
   - **Determinar se a linguagem gerada por uma GLC é um subconjunto da linguagem gerada por outra GLC;**

- **etc.**

9. (1 ponto) Conceitue:
    a. Classe P;

    Conjunto das linguagens que podem ser decididas por uma Máquina de Turing determinística de tempo polinomial.

    b. Classe NP;

    Conjunto das linguagens que podem ser decididas por uma Máquina de Turing não-determinística de tempo polinomial.

    c. Redução em tempo polinomial;

    Redução que mapeia cadeias de uma linguagem A em cadeias de uma linguagem B. Ela ocorre em tempo que é uma função polinomial do comprimento da cadeia de A.

    d. Problema NP-completo.

    Problema pertencente à NP para o qual existem reduções de tempo polinomial de todas os problemas pertencentes à NP para ele.

10. (0,5 ponto) De que maneira uma redução de tempo polinomial deve ser usada para provar que um certo problema é solucionável em tempo polinomial?

    Apresentando-se uma redução de tempo polinomial do problema em questão para um segundo problema solucionável em tempo polinomial.

11. (0,5 ponto) Em que consiste a questão P=NP?

    Consiste na determinação da relação que existe entre as classes P e NP. As duas possibilidades são (i) a classe P é igual à classe NP, ou (ii) a classe P é um subconjunto próprio da classe NP. A resposta não é conhecida.

12. (1 ponto) Se algum dia for provado que existe algum problema pertencente à NP que não pertence à P, o que se poderá concluir em relação à todos os problemas NP-completos? Eles pertencem à P, à NP, ou não será possível afirmar nada de forma genérica? Justifique a sua resposta.

    Se isso for provado um dia, também ficará provado que todos os problemas NP-completos pertencem à NP-P. Se assim não fosse, uma solução eficiente para um problema NP-completo poderia ser usada para obter uma solução eficiente para o problema em questão, o que seria uma contradição.