

Exame de Qualificação 08/07/2014

Leila Ribeiro e Luciana S. Buriol

March 25, 2017

Teoria da Computação

Perguntas

1. (2.5 pontos) Imagine que se quer provar que uma propriedade P é válida para a classe de funções computáveis. Faça esboços de como poderiam ser as provas desta propriedade usando Máquinas de Turing e usando Funções Parciais Recursivas, justificando a validade de cada prova. Defina critérios e faça uma análise comparativa das duas provas.
 - FAZER
2. (2.5 pontos) Disserte sobre *problemas NP-Completo*. Além das definições, explicações e exemplos que você julgar relevantes, sua resposta deve necessariamente explicar 2 formas diferentes de se provar que um dado problema é NP-Completo e também uma justificativa da importância desta classe de problemas. Note que a sua escolha do que é relevante sobre este assunto será considerada na nota desta questão.
 - A classe de problemas NP-Completo tem como característica problemas de decisão em que a melhor solução conhecida leva tempo exponencial em relação ao tamanho da entrada, e que ao mesmo tempo possui um tempo de verificação ou certificação da solução em tempo polinomial. Uma das características mais importantes desta classe de problemas é o fato de que todos os problemas são redutíveis entre si em tempo polinomial fazendo uso de uma Máquina de Turing determinística. Assim, caso uma solução polinomial seja obtida para um dos problemas desta classe, isso implicaria na solução de todos os demais problemas NP-Completo.
 - A definição mais formal para um problema NP-Completo é a de que este problema pertence ao mesmo tempo ao conjunto de problemas NP e NP-Hard. Existem duas formas de provar que um problema R é NP-Completo:
 - (a) Reduzir o problema R em tempo polinomial utilizando uma Máquina de Turing determinística para um problema S que pertence ao conjunto de problemas NP-Completo. Seguido de uma redução, também em tempo polinomial por uma MTD, de S para R . Mostrando assim que ambos os problemas são redutíveis entre si, logo, ambos pertencem a classe de problemas NP-Completo. Para que esta prova seja possível é necessário a existência de pelo menos um problema conhecido por estar em NP-Completo (Teorema de Cook-Levin).
 - (b) Primeiramente é necessário mostrar que o problema R pertence ao conjunto de problemas NP, ou seja, que é possível obter uma solução em tempo polinomial para o problema utilizando uma Máquina de Turing Não-Determinística, ou tempo polinomial para verificação da corretude da solução por uma Máquina de Turing Determinística. E por fim é preciso mostrar que R é NP-Hard, para isso todos os demais problemas em NP precisam ser redutíveis em tempo polinomial por uma MTD para o problema R . Logo, se um problema pertence ao conjunto de problemas NP-Hard e NP, então ele é NP-Completo. O problema

da *Boolean Satisfiability* foi identificado de forma independente como sendo NP-Completo por Cook e Levin, estabelecendo assim um problema raiz para esta classe e permitindo que outros inúmeros problemas fossem também colocados nesta categoria.

- Esta classe de problemas também tem um papel importante na questão $P = NP?$. Os problemas identificados como NP-Completo são considerados os problemas mais difíceis em NP, no sentido de que existem reduções em tempo polinomial de cada um dos problemas em NP para um problema NP-Completo. Ou seja, no momento em que uma solução obtida para um determinado problema pertencente ao conjunto NP-Completo apresente tempo polinomial, demonstrando assim que este problema está em P, todos os demais problemas em NP poderão também ser resolvidos em tempo polinomial, provado que $P = NP$ é verdadeiro. Porém, ainda não existe uma resposta para essa pergunta, e a dúvida para decidir se P é um subconjunto de NP ou se ambos contém exatamente o mesmo conjunto de problemas persiste, sendo conhecido como um dos problemas mais difíceis na Ciência da Computação.