Exame de Qualificação 04/03/2015

Álvaro Freitas Moreira e Luciana S. Buriol

March 27, 2017

Teoria da Computação

Perguntas

1. **(2.5 pontos) Disserte sobre os conjuntos de problemas NP, NP-Difíceis e NP-Completos. Desenhe diagramas de Venn mostrando a relação entre estes três conjuntos para o caso de P = NP e P $\neq$ NP.**

   Escrever.

2. **(2.5 pontos) Disserte sobre os seguintes aspectos da *Teoria da Computabilidade*.**

   (a) **Qual é o seu principal foco de estudo;**

   - O principal foco da Teoria da Computabilidade está no estudo de problemas de decisão precisamente definidos, os quais estão associados à um conjunto de entradas, cuja solução é válida ou não para qualquer valor pertencente a este conjunto, sendo que se trata de um conjunto que apresenta infinitas possibilidades de entradas válidas. Estes problemas de decisão apresentam como saída *Sim* ou *Não* (*true* ou *false*) caso o programa tenha obtido uma decisão válida ou não para as informações de entrada.

   (b) **Exemplos dos principais resultados (teoremas, teses, etc);**

   - **Teorema de Rice**: Nenhuma propriedade interessante que diz respeito a computação é computável. Por interessante podemos definir propriedades que se referem a função que está sendo computada pela MT ou pelo algoritmo, e não o que diz respeito à sua representação ou forma. Um exemplo de propriedade da representação é o número de linhas de um programa, ou o número de estados de uma MT (computável), enquanto um exemplo de propriedade interessante é saber se a MT ou função sempre irá retornar números pares ou se dois programas diferentes representam a mesma função (não-computável).
   - **Tese de Church-Turing**: A Tese de Church-Turing iguala a noção subjetiva de efetivamente computável com a definição formal de computável por uma Máquina de Turing. Ou seja, mesmo nunca sendo provada e ainda assim sendo aceita pela comunidade, esta tese estabelece que se uma função pode ser computada por uma Máquina de Turing então ela é efetivamente computável.

   (c) **Relevância para a Computação desses resultados principais.**

   - O Teorema de Rice é importante para áreas como a Engenharia de Software, uma vez que define que as propriedades acerta da computação são não-computáveis, sendo assim não é possível desenvolver um aplicativo ou um plug-in que, por exemplo, seja capaz de verificar se um código irá ser executado com sucesso para qualquer entrada válida.
   - Já a Tese de Church-Turing é extremamente importante uma vez que estabele os limites entre o que é computável e o que não é. Sendo assim, uma vez definido um problema que

não pode ser computado por um MT, este é tido como não-computável de qualquer maneira. A partir desse problema conhecido como não tendo uma solução computável, outros podem ser reduzidos à ele, permitindo então agrupar e identificar outros problemas que não possuem uma solução efetivamente computável.

3. **(2.5 pontos) Disserte sobre os seguintes aspectos da *Teoria da Complexidade*.**

   (a) **Qual é o seu principal foco de estudo;**

   - O principal foco de estudo da Teoria da Complexidade é a complexidade intrinseca de problemas computacionais. Dentre os objetivos desta área pode-se mencionar a descoberta da complexidade concreta de um problema, ou seja, quanto tempo e espaço um problema leva para resolvido, e também a identificação de conexões entre as diferentes complexidades apresentadas pelos problemas. Este segundo objetivo, por sua vez, levou a descoberta e definição de diferentes classes de complexidade, tais como as de problemas P, NP, NP-Completos, dentre outros.

   (b) **Exemplos dos principais resultados (teoremas, teses, etc);**

   - **Teorema de Cook-Levin**: estabelece que o problema da satisfatibilidade booleana (SAT) por expressões na forma normal conjuntiva (CNF) é NP-Completo. O problema SAT foi o primeiro da história da computação a ser provado como NP-Completo. A prova foi realizada de forma independente por Cook e Levin, os quais provaram que SAT está em NP e propuseram um problema geral em NP (**representando todos os problemas em NP**) que foi reduzido a SAT em tempo polinomial. Dessa forma, ambos provaram que todo problema em NP pode ser reduzido a SAT em tempo polinomial.

   (c) **Relevância para a Computação desses resultados principais;**

   - O Teorema de Cook-Levin é de extram importância para a Cincência da Computação por vários aspectos. Um dos principais é o fato de ser o percursor das discussões acerca da NP-Completude, conceito que não existia à época. Outro fator importantíssimo é que, por provar o primeiro NP-Completo, o Teorema facilitou muito a inclusão de outros problemas a essa classe. Seguem duas possibilidades de provar que um novo problema é NP-Completo: (1) provar que o problema é tanto NP quanto NP-Hard. Para isso, prove uma possível solução pode ser verificada em tempo polinomial (*prova de que está em NP*), e também prove que o problema pode ser reduzido a um problema NP-Completo já conhecido (*prova indireta que o problema é NP-Hard*) ou então mostre que qualquer problema que já pertence ao conjunto de problemas em NP pode ser reduzido à este problema (*prova direta que o problema é NP-Hard*); (2) utilize a inter-redutibilidade entre os problemas NP-Completos. Para isso escolha um problema NP-Completo X já conhecido, reduza o novo problema a X e, depois, reduza X ao novo problema. Com isso estará provado que o novo problema não é pior do que X, quanto ele não pode ser melhor que X, logo o novo problema também NP-Completo.

   (d) **Problemas em aberto.**

   - **P vs. NP**?: esta é considerada a questão aberta central na Teoria da Complexidade e é também uma questão importante para a ciência como um todo. A questão P vs. NP pode ser vista como uma dúvida relacionada à complexidade de um determinado problema. Sabemos que é possível resolver um problema que pertence a classe P em tempo polinomial, e também sabemos que é possível verificar se uma solução em tempo polinomial para um problema que pertence a classe NP. Caso a firmação seja verdadeira P $=$ NP, isso significa que todos os problemas verificáveis em tempo polinomial, também podem ser resolvidos em tempo polinomial, caso seja falsa P $\neq$ NP, então problemas que estão em NP apenas podem ser resolvidos em tempo super-polinomial. Supondo que P $=$ NP, o impacto no mundo seria enorme, algoritmos de criptografia como SSL, RSA e PGP (que pertencem a NP) possuiriam uma solução em tempo polinomial.