

Exame de Qualificação 04/03/2015

Álvaro Freitas Moreira e Marcus Ritt

March 26, 2017

Teoria da Computação

Perguntas

1. (2.5 pontos) Um professor gostaria de automatizar a correção dos exercícios de programação Python dos alunos, construindo um programa que implemente a função **PROG-CORRETO?**, definida assim:

Entrada: Dois programas escritos em Python, um representando a solução (*sol*) e outro o programa submetido pelo aluno (*/sub*).

Saída: *Verdadeiro*, se o programa *sub* implementa a mesma função que o programa *sol*, ou seja, se para todas as entradas para as quais *sol* produz um valor de saída, *sub* produz o mesmo valor, e para todas as entradas para as quais *sol* não produz saída, *sub* também não produz saída. A saída deve ser *Falso* caso contrário.

Responda as seguintes questões:

- (a) Explique o que é uma "*função computável*".
 - De acordo com a Tese de Church-Turing, uma função é dita efetivamente computável, se e somente se, ela for Turing-computável. Isto é, para que uma função seja computável, deve existir uma máquina de Turing capaz de representar esta mesma função.
- (b) A função **PROG-CORRETO?** é computável? Prove.
 - A afirmação é falsa. De acordo com o Teorema de Rice, nada de interessante ou não-trivial a respeito da computação é computável. Neste caso, o termo *interessante* diz respeito ao comportamento e a semântica de uma determinada função para todo o conjunto possível de entradas. Logo, é impossível construir um programa para avaliar se o comportamento das funções *sol* e *sub* são idênticas.
- (c) Transforme a definição de **PROG-CORRETO?** em **PROG-CORRETO?mod** da seguinte forma:
 - Se sua resposta na questão b) foi **SIM**, **PROG-CORRETO?mod** deve ser uma função mais geral que **PROG-CORRETO?** que não seja computável.
 - Se sua resposta na questão b) foi **NÃO**, **PROG-CORRETO?mod** deve ser uma função mais restrita que **PROG-CORRETO?** que seja computável.

Justifique por que **PROG-CORRETO?mod** é computável ou não computável.

- Para que a definição de **PROG-CORRETO?** seja computável é necessário realizar testes em um conjunto definido de entradas. Por exemplo, **PROG-CORRETO?mod** deve verificar se para um conjunto definido de entradas ($[1,2,3,4,5]$) os programas *sol* e *sub* produzem o mesmo resultado, uma vez que as saídas de ambos os programas podem ser comparado após a sua execução, e assumindo que *sol* e *sul* são equivalentes se os resultados obtidos para

as entradas informadas são idênticos, a função então é tida como computável. **Melhorar resposta e formatar. Esclarecer por que é computável.**

2. (1 ponto) Disserte sobre as principais classes de complexidade e explique a importância de se demonstrar a classe de complexidade de um problema.

- **Escrever.**

3. (1.5 pontos) Diga se as afirmações a seguir são verdadeiras ou falsas e justifique suas respostas.

- (a) Se um problema pertence à classe P , sempre existe um algoritmo com complexidade $O(2^n)$ que o resolve.

- Verdadeiro. Se um problema pertence a classe P , então existe um algoritmo que é executado em tempo polinomial capaz de resolver este problema. **Qualquer problema da classe P pode ser reduzido a um problema de uma ordem superior, apresentando tempo de execução exponencial $O(2^n)$.**

- (b) Se $P = NP$ então a lógica de primeira ordem é decidível.

- Falso. Se $P = NP$ então todos os problemas da NP-Completo poderiam ser resolvidos em tempo polinomial. **Verificar resposta.**

- (c) Se L é NP-Hard, L é polinomialmente redutível a L' e L' pertence a P então $P \neq NP$.

- Falso. $P \neq NP$, então problemas NP-Hard não podem ser redutíveis a um problema da classe polinomial. **Verificar resposta.**