Teoria da Computação
Fundamentos da Computação

Simão Melo de Sousa

Exercício 1 *Uma técnica popular para a demonstração de indecidibilidade de problemas,* a técnica da redução, *consiste em exibir uma transformação do problema estudado para um problema conhecido por ser indecidível.*

Se encontrar uma solução para um problema A pode se transformar na (ou pode equivaler numa) procura duma solução para o um problema B, então se B não tem solução algorítmica de certeza que A também não tem. O problema A é assim igualmente indecidível.

Neste contexto, é importante que a referida transformação seja ela própria decidível (tem de existir um algoritmo que a efectue).

Diga porque esta última condição é essencial. Sugestão: digo por exemplo o que aconteceria (em termos de conclusão por tirar sobre o problema A) se a transformação não pudesse ser um algoritmo. **Resposta** □

Exercício 2 *O problema da paragem questiona se existe um método algorítmico que possa estabelecer sem falha se qualquer programa que lhe for fornecido como parâmetro termina ou não. Foi demonstrado que tal método não pode existir. No entanto foi introduzido nesta disciplina um método baseado no princípio da indução bem fundada que permite demonstrar a terminação. Comente e resolve este aparente paradoxo.*

Resposta □

Exercício 3 *As máquinas de Turing são mecanismos computacionais teóricos, cite pelo menos um argumento que impossibilite a sua implementação directa na forma duma arquitectura física (processador, memoria, etc.).*

Resposta □

Exercício 4

Como caracterizar a complexidade em tempo da solução dum problema indecidível?

Resposta

□

Exercício 5 *Que impacto terá a descoberta dum algoritmo que não se pode expressar com a ajuda duma máquina de Turing?*

Resposta

□

Exercício 6 *O problema da paragem é indecidível, no entanto sabemos decidir (demonstrar) se os peudocódigos seguintes:*

```
while (true) i++;
while (i>0) i-;
```

terminam ou não (de facto o primeiro não termina, o segundo sim). Outra exemplo é sabermos demonstrar mecanicamente que qualquer recursão estrutural termina. Explique esta aparente contradição.

Resposta

□

Exercício 7 *As noções de indecidibilidade e de não computabilidade estão ligadas a noção de conjunto infinito. Explique esta relação.*

Resposta □

Exercício 8 *Na introdução desta disciplina afirmamos que iríamos definir e estudar os limites do que é possível resolver por computador. Ao longo do semestre concentramo-nos exclusivamente sobre problemas de linguagens, sobre mecanismos geradores e sobre o reconhecimento de linguagens. Explique porque esta discrepância é só aparente, ou seja, apresente a relação entre os mecanismos computacionais e as linguagens formais.* **Resposta** □

Exercício 9 *Classicamente associamos a noção de totalidade (da função sobre o domínio de interesse) à noção de computabilidade/decidibilidade. Um problema é decidível se existe uma processo/função total que calcule a solução ao problema (isto é, que converge para a solução qualquer que seja a instância* *considerada). Neste cenário, explique e contextualize a noção de parcialidade (i.e. função parcial sobre o domínio de interesse). Que representa ela quando tentamos qualificar a decidibilidade (ou não) dum problema?* **Resposta** □

Exercício 10

Explique porque o programa de Hilbert obrigou a uma definição rigorosa do conceito de procedimento efectivo. **Resposta** □

Exercício 11

Define o conceito de decicibilidade de uma teoria; Define também o conceito de completude. Discute a relação e as diferenças entre as duas noções.

Resposta □

Exercício 12

1. *Explique o que enuncia a Tese de Church-Turing e a sua importância nas ciências e na Informática em particular.*

2. *Uma tese não e um teorema. Explique a diferença. Em particular explique as implicações deste facto sobre a tese de Chuch-Turing (é universal e intemporal? é sólida?).*

Resposta □