Aula 13: Computabilidade

Prof. Lucio A. Rocha

Engenharia de Computação
Universidade Tecnológica Federal do Paraná, UTFPR
Campus Apucarana, Brasil

2º semestre / 2023

Sumário

Seção 1

Computabilidade

Computabilidade

- Computabilidade é a investigação da existência ou não de algoritmos que solucionam determinada classe de problemas.
- Pesquisar os limites da computabilidade é estudar os limites do que pode ser efetivamente implementado por um computador.
- Evitar a pesquisa de soluções não-computáveis.

Computabilidade

Função computável

Def.: Uma função é computável se existe um algoritmo que calcula os valores de saída a partir dos valores de entrada.

- Ex.: $\pi = \frac{comprimento}{diametro}$
- Ex.: $y = \sqrt{x}$

Computabilidade

Função não-computável

Def.: Se não existe algoritmo que resolva um problema, então esse problema não é computável (Gödel).

- Ex.: Problema da parada: dada uma entrada P qualquer, a função reconhece (informa) que há um loop.
- Ex.: Problema da totalidade: há uma função que decide (implementa a ação) que o programa sempre pára, para qualquer entrada.

Computabilidade

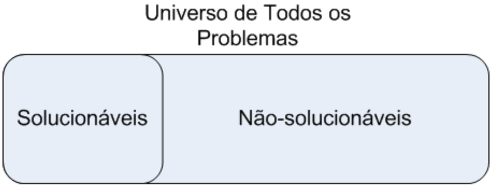

Figura: Classes de Problemas.

Computabilidade

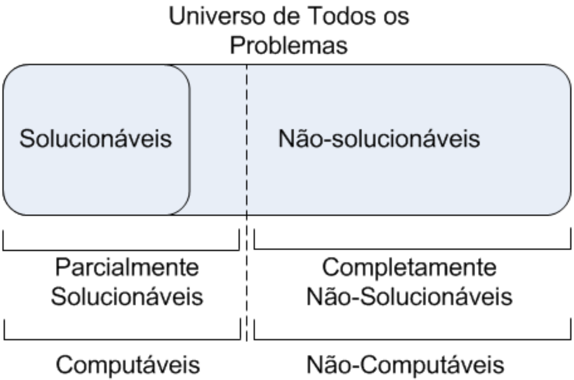

Figura: Classes de Problemas.

Computabilidade

Problema Solucionável

Def.: Um problema é solucionável se existe um algoritmo (Máquina Universal) que soluciona o problema.
O algoritmo sempre pára para qualquer entrada em um estado de ACEITA ou REJEITA.

Problema Não-Solucionável

Def.: Um problema é <u>não-solucionável</u> se <u>não</u> existe um algoritmo (Máquina Universal) que soluciona o problema e
que sempre pára para qualquer entrada.

Computabilidade

Problema Parcialmente Solucionável (ou Computável)

Def.: Um problema é parcialmente solucionável (ou computável) se existe um algoritmo (Máquina Universal) que soluciona o problema, e que pára quando a resposta é ACEITA.
Porém, quando a resposta não é aceita, o algoritmo pode parar (REJEITA) ou entrar em loop.

Problema Não-Computável

Def.: Um problema é completamente insolúvel (ou não-computável) se não existe um algoritmo (Máquina Universal) que soluciona o problema, e que pára quando a resposta é ACEITA.

Computabilidade

Problema Não-Solucionável

Def.: Um problema é não-solucionável, completamente insolúvel ou não-computável se não existe um algoritmo (Máquina Universal) que soluciona o problema, e que pára quando a resposta é ACEITA.

Seção 2

Tese de Church-Turing

Tese de Church-Turing

Tese de Church-Turing

Def.: Se uma função é efetivamente computável então ela é computável por meio de uma Máquina de Turing.

- Na prática:
  - Se um problema tem solução então ele é computável com uma MT.
  - Ou seja: existe um algoritmo que expressa a solução do problema.
  - <u>Todo</u> algoritmo pode ser expresso por uma MT.

Seção 3

Redutibilidade

Redutibilidade

- Investigar a solução de um problema, a partir de um caso particular de outro problema.
  1. Sejam A e B dois problemas de decisão. Suponha que seja possível modificar ("reduzir") o problema A para que este se torne um caso particular do problema B.
  2. Se A não é solucionável (i.e.,não-computável) então B também não é solucionável (i.e.,não-computável).
  3. Se B é solucionável (i.e.,parcialmente solucionável) então como A é um caso particular de B, o problema A também é solucionável (i.e., parcialmente solucionável).

Redutibilidade

- Exemplo aplicado na Lógica de Predicados:
- $\forall x(P_x \wedge Q_x) \vdash \forall x P_x \wedge \forall x Q_x$

1.	$\forall x(P_x \wedge Q_x)$	hip.
2.	$P_a \wedge Q_a$	EU 1
3.	P_a	SP 2
4.	Q_a	SP 2
5.	$\forall x P_x$	IU 3
6.	$\forall x Q_x$	IU 4
7.	$\forall x P_x \wedge \forall x Q_x$	Conj. 5,6

Redutibilidade

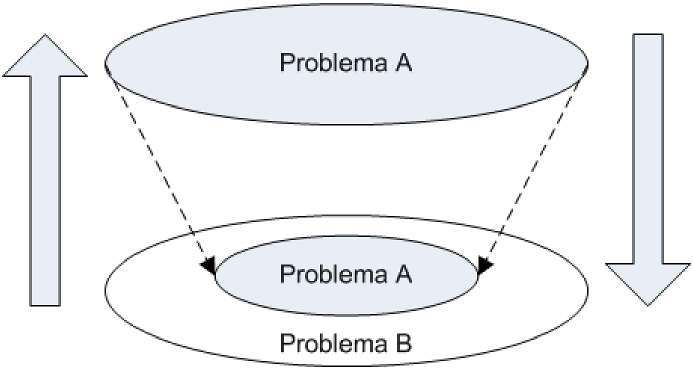

Figura: Redutibilidade.

Seção 4

Decidibilidade

Decidibilidade

- Decidibilidade é a verificação se a função associada a um problema de decisão é ou não computável em uma Máquina Universal.
- Um problema é decidível se existe uma MT que o resolve.
- Um Problema é indecidível (insolúvel) se não pode ser expresso em Máquina Universal.
- Decidibilidade: Dado um programa P para uma Máquina Universal M, decidir se a função computada $< P, M >$ é total, ou seja, se a computação é finita.

Decidibilidade

- Problema de decisão: é uma pergunta sobre um problema com uma resposta binária: sim ou não.
- Problema de decisão: determinar se um elemento pertence ou não a um conjunto.
  - Ex.: *7 é primo?*
- O método utilizado para resolver um problema de decisão é o algoritmo.

Decidibilidade

Problema de decisão decidível

Def.: Dado um conjunto A, o problema de decisão é decidível (ou solucionável) se A é um conjunto recursivo.

Decidibilidade

- Conjunto recursivo (computável ou decidível): é um conjunto de números Naturais sobre o qual existe um algoritmo que termina após um tempo finito e decide (informa) se o número pertence ou não ao conjunto.
- Ex.: $A = \{$ Conjunto dos números primos menores que 10 $\}$

Decidibilidade

- Conjunto recursivamente enumerável (ou semidecidível): é um conjunto de números Naturais sobre o qual existe um algoritmo que termina após um tempo finito e decide (informa) apenas se o elemento pertence ao conjunto.
  Porém, o algoritmo pode não dar uma resposta (mas não uma resposta errada) caso o elemento não pertença ao conjunto.
- Todo conjunto recursivo é recursivamente enumerável.
- Ex.: $A = \{$ Conjunto dos números primos menores que 10 $\}$
- Ex.: $B = \{$ Conjunto dos números primos $\}$

Seção 5

Problema da Parada

Problema da Parada

Problema da Parada

Def.: Dado um programa e uma entrada quaisquer, não existe um algoritmo genérico capaz de verificar se o programa vai parar ou não para a entrada.

- O problema da parada é um problema de decisão que, dado um programa P e uma entrada finita, decide (verifica) se o programa termina de executar, ou entra em loop.

Seção 6

Intratabilidade

Intratabilidade

Problemas intratáveis

Problemas intratáveis são aqueles que, teoricamente, podem ser resolvidos, mas que, na prática, levam muito tempo para que as suas soluções sejam úteis.