I.2 – Introdução a Teoria da Computação

O que é ?

- Fundamento da Ciência da Computação
- Tratamento Matemático da Ciência da Computação
- Estudo Matemático da Transformação da Informação

Qual sua importância?

Guia : “Que problemas podem ser efetivamente computáveis, como e com que complexidade”

Classificação dos problemas:

- **Não-Computáveis**
- **Computáveis**
    - **Indecidíveis**
    - **Decidíveis**
        - **Intratáveis**
        - **Tratáveis**

Exemplos de problemas:

- Computabilidade e Decidibilidade
- Expressividade de Máquinas e Modelos
- Equivalência entre Programas
- Complexidade Computacional
- Semântica e Correção de Programas

Problemas Computacionais

X

Teoria das Linguagens Formais

➔Todo problema computacional pode ser tratado como um problema de LINGUAGENS !!!

Problema Computacional

↓ Pode ser solucionado por

Programa / Função P : Entradas X Saídas

↓ Possui ...

Algoritmo de DECISÃO

↓

Dp (Entradas, Saída) ➔ { True, False }

➔Exemplo:

Dados **p** e **q** inteiros, $\mathbf{q = p^2}$?

∴ Se para qualquer Entrada e para qualquer Saída submetidos a P, Dp sempre responde True ou False,
<u>então</u> o problema em questão é DECIDÍVEL,
<u>senão</u> o problema é INDECIDÍVEL

Correlacionando Problemas Computacionais com Problemas de Linguagens

- Entradas e Saídas podem sempre serem representadas por sequências de caracteres (ex. usando o alfabeto unário { a }, podemos representar : 1 por "a", 2 por "aa", ... , N por a^N);
- Uma Linguagem nada mais é que um conjunto de sentenças (sequências) formadas sobre um alfabeto;
- Pares (Entrada, Saída) podem ser vistos como sequências (ou sentenças) pertencentes a uma LINGUAGEM

➔Assim, se uma sequência particular pertence à linguagem formada pelas soluções de um problema qualquer, ela será uma solução do problema em questão!!!

➔ No exemplo dado, a LINGUAGEM pode ser enunciada como sendo **L = { (a^N, a^{N2}), onde "a" é o símbolo do alfabeto e N representa um inteiro positivo qualquer}**

∴ (2, 4) e (5, 25), representam instâncias True, enquanto que (1, 3) e (3, 17) representam instâncias False para o algoritmo de decisão relativo ao problema dado.

Conclusão : Como o algoritmo de decisão relativo ao problema em questão, sempre dará uma resposta (**True ou False**), para qualquer par de inteiros fornecidos, o problema é **DECIDÍVEL**!

➔Ex.: o problema abaixo é decidível?

Dado um inteiro positivo N, Existe X, Y, Z (diferentes de 0), tal que $X^N + Y^N = Z^N$???

Teoria da Computação
X
Teoria das Linguagens Formais

Definição de Teoria da Computação sob a ótica da Teoria das Linguagens Formais:

Conjunto de Modelos Formais (autômatos e gramáticas, p. ex.), que juntamente com suas propriedades (decidibilidade, equivalência e complexidade), fundamentam a Ciência da Computação.

Conceitos e Propósitos Fundamentais da Teoria da Computação

PROCEDURE X ALGORITMO

Procedure : Sequência finita de passos, executáveis mecanicamente de forma discreta.

Algoritmo : É uma procedure que, independentemente das entradas, possui parada garantida.

Exemplos:

- **Determinar se I ≥ 1 é um número primo**
- **Determinar se existe um número perfeito > I**
- **Determinar se um programa está sintaticamente correto**
- **Determinar se um programa qualquer entrará em loop para uma entrada qualquer (Halting Problem)**

Conj. Recursivos e Recursivamente Enumeráveis

X

Algoritmos e Procedures

X

Problemas Decidíveis e Problemas Indecidíveis

Propósitos da Teoria da Computação

(ou : <u>Modelos</u> que formalizam a noção do que é do que não é <u>efetivamente computável</u>)

Máquinas de Turing (Turing, 1936)

→ Tese de CHURCH – Todo processo efetivo pode ser realizado por uma máquina de turing)

Gramáticas (Chomsky, 1959)

→Tipos 0, 1, 2 e 3

Algoritmos de Markov (Markov, 1951)

→ Sistemas de regras de produção

→ Processamento de strings

λ-Calculus (Church, 1936)

→ Método para especificação de funções

→ Influenciou programação funcional

Sistemas de POST (Emil Post, 1936)

→ Formalização de sistemas de re-escrita

→ Lógica formal e sistemas especialistas

Funções Recursivas (Kleene, 1936)

→Método para definição de funções a partir de um conjunto de equações

exemplo : $X^Y = 1, \quad \text{se } y = 0$

$\quad\quad\quad\quad = X \cdot X^{Y-1}, \text{ se } y > 0$

I.2.1 – Introdução à Teoria das Linguagens Formais

- **O que é LINGUAGEM FORMAL ?**
- **O que é LINGUAGEM ?**
  - Forma de comunicação
  - Conjunto de símbolos + conjunto de regras
  - Exemplos: L. Máquina, PASCAL, Português, ...

I.2.1.1 – Conceitos Básicos

- Alfabeto :
  - Conjunto finito e não vazio de símbolos
- Sentença:
  - Sequência de símbolos de um alfabeto
- Tamanho de uma sentença:
  - Quantidade de símbolos de uma sentença
- Sentença vazia:
  - denotada por ε, é uma sentença de tamanho 0
- Potência de uma sentença:
  - exemplo: $a^3 = aaa$
- Fechamento de um alfabeto:
  - Reflexivo : V^*
  - Transitivo (ou Positivo) : V^+

I.2.1.2 – Linguagens e suas Representações

- **Linguagem :**

$$L \subseteq V^*$$

- **Formas de Representação:**
  - Enumeração
  - Sistemas Geradores
  - Sistemas Reconhecedores
- **Linguagens Formais**
  - Dispositivos / Modelos matemáticos
- **Linguagens Recursivas:**
  - Algoritmos
- **Linguagens Recursivamente Enumeráveis**
  - Procedures
- **Teoria das Linguagens Formais:**
  - Estudo dos modelos matemáticos que possibilitam a especificação, o reconhecimento, a classificação, as propriedades e o interrelacionamento entre linguagens.

- **Importância da Teoria das Linguagens:**
  - Apóia aspectos básicos da Teoria da Computação:
    - Decidibilidade, Computabilidade, Complexidade
  - Fundamenta Aplicações Computacionais:
    - Processamento de Linguagens (esp. / impl.),
    - Rec.de Padrões, Modelagem de Sistemas, ...