Revisão

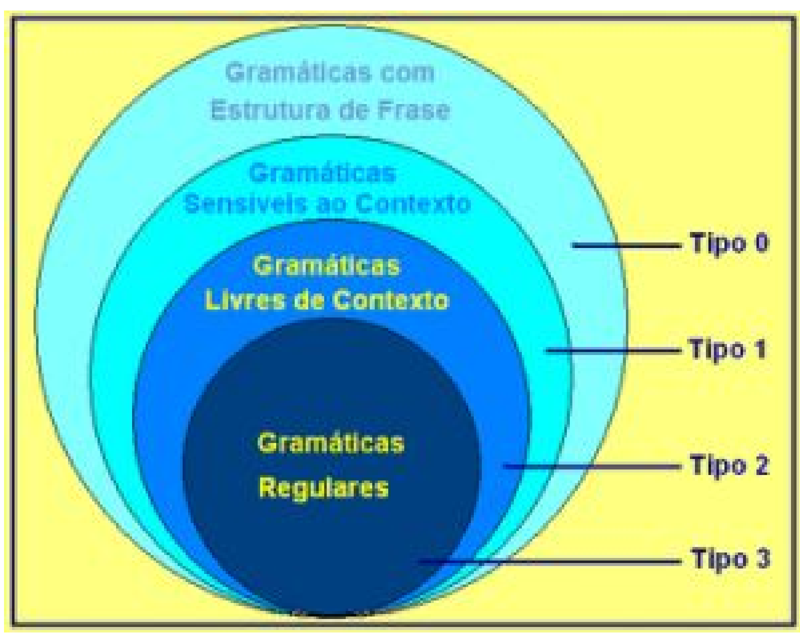
http://www.din.uem.br/yandre/TC/lista1-resp.pdf -> helpzinho topper

Hierarquia de Chomsky

É a classificação de gramáticas formais descrita em 1959 pelo lingüista Noam Chomsky. Esta classificação possui 4 níveis (Descritos na figura ao lado), sendo que os dois últimos níveis (os níveis 2 e 3) são amplamente utilizados na descrição de linguagem de programação e na implementação de interpretadores e compiladores. Mais especificamente, o nível 2 é utilizado em análise sintática (computação) e o nível 3 em análise léxica.

A classificação das gramáticas começa pelo tipo 0, com maior nível de liberdade em suas regras, e aumentam as restrições até o tipo 3. Cada nível é um super conjunto do próximo. Logo, uma gramática de tipo n é conseqüentemente uma linguagem de tipo n-1.

- Linguagem tipo 3: Regular
    - Reconhecida por um autômato finito
    - Representada por uma expressão regular
- Linguagem tipo 2: Livre de contexto
    - Reconhecida por um autômato de pilha
- Linguagem tipo 1: sensível ao contexto
- Linguagem tipo 0: Irrestrita
    - Máquina de Turing



Prova 1

Máquina de Turing

A Máquina de Turing é um dispositivo teórico conhecido como máquina universal, que foi concebido pelo matemático britânico Alan Turing (1912-1954), muitos anos antes de existirem os modernos computadores digitais (o artigo de referência foi publicado em 1936). Num sentido preciso, é um modelo abstrato de um computador, que se restringe apenas aos aspectos lógicos do seu funcionamento (memória, estados e transições) e não à sua implementação física. Numa máquina de Turing pode-se modelar qualquer computador digital.

Uma máquina de Turing consiste em:

- Uma fita que é dividida em células, uma adjacente à outra. Cada célula contém um símbolo de algum alfabeto finito. O alfabeto contém um símbolo especial branco (aqui escrito como ¬) e um ou mais símbolos adicionais. Assume-se que a fita é arbitrariamente extensível para a esquerda e para a direita, isto é, a máquina de Turing possui tanta fita quanto é necessário para a computação. Assume-se também que células que ainda não foram escritas estão preenchidas com o símbolo branco.

- Um cabeçote, que pode ler e escrever símbolos na fita e mover-se para a esquerda e para a direita.

- Um registrador de estados, que armazena o estado da máquina de Turing. O número de estados diferentes é sempre finito e há um estado especial denominado estado inicial com o qual o registrador de estado é inicializado.

- Uma tabela de ação (ou função de transição) que diz à máquina que símbolo escrever, como mover o cabeçote (← para esquerda e → para direita) e qual será seu novo estado, dados o símbolo que ele acabou de ler na fita e o estado em que se encontra. Se não houver entrada alguma na tabela para a combinação atual de símbolo e estado então a máquina pára.

Note que cada parte da máquina é finita; é sua quantidade de fita potencialmente ilimitada que dá uma quantidade ilimitada de espaço de armazenamento.

Funcionamento

- A Máquina de Turing deve assumir sempre em um estado, pertencente à um conjunto finito de estados;

- O processamento de uma Máquina de Turing começa sempre em um estado especial, chamado estado inicial;

- Inicialmente a primeira célula da fita é preenchida com "⟨", que indica o início da mesma;

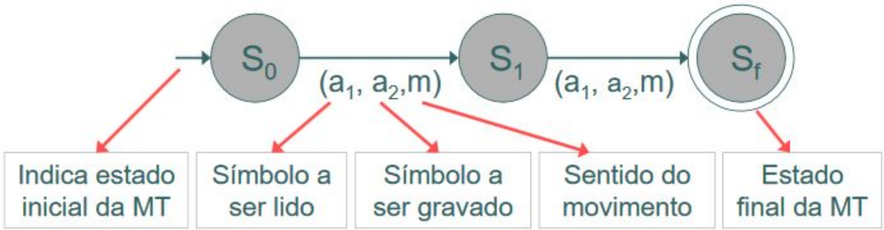
- A cabeça de leitura é posicionada inicialmente na segunda célula da fita, a célula seguinte a "〈";
- As células em branco, que não fazem parte da palavra a ser processada, são preenchidas com o símbolo "β";
- O processamento em uma MT consiste em uma sequência de passos que consistem em:
    - Observar o símbolo corrente da fita (aquele em que o cabeçote está posicionado);
    - Escrever um símbolo nesta célula da fita;
    - Mover o cabeçote para a esquerda ou direita;
    - Mudar o estado corrente;
    - A ação exata a ser executada é determinada por um programa (função de transição) que comunica à unidade de controle o que deve ser feito com base na configuração (estado + símbolo corrente da fita) em que a Máquina de Turing se encontra.
- O processamento termina quando a máquina atinge uma configuração para a qual não existe função prevista, neste caso:
    - Se a máquina estiver com um estado final ativo, a palavra de entrada é aceita;
    - Se o estado ativo não for final, a palavra de entrada não é aceita;
    - Se a máquina não parar (looping), a entrada de entrada não é aceita.

A Máquina de Turing pode ser empregada como modelo de natureza reconhecedora ou transdutora:

- Reconhecedora: processa a palavra de entrada aceitando-a ou rejeitando-a. Neste caso, o conjunto de palavras aceitas corresponde à linguagem descrita pela MT;
- Transdutora: máquina para computar uma função. Aplica uma função sobre o conteúdo inicial da fita e o resultado produzido é lançado na própria fita.

Representação de Máquina de Turing por diagrama:

- Semelhante à representação de Autômato finito determinístico;
- Os rótulos das setas, que representam as funções de transição são de acordo com a seguinte legenda:



$L = \{\, w \mid w \in \{0,\ 1\}^{*}\ e\ w = \ w^{R}\ \}$

$L = \{\, \varepsilon,\ 0,\ 1,\ 00,\ 11,\ 000,\ 010,\ 101,\ 111,\ \ldots\}$

- Q0
  - Se for 0 -> Assina M. Vai para a Direita. Q1.
  - Se for 1 -> Assina M. Vai para a Direita. Q4.
  - Se for ε ou M -> Aceita.
- Q1
  - Se for 0 ou 1 -> Vai para a Direita. Q1.
  - Se for ε ou M -> Vai para a Esquerda. Q2.
- Q2
  - Se for 0 -> Assina M. Vai para a Esquerda. Q3.
  - Se for M -> Vai para a Esquerda. Aceita.
- Q3
  - Se for 0 ou 1 -> Vai para a Esquerda. Q3.
  - Se for M -> Vai para a Direita. Q0.
- Q4
  - Se for 0 ou 1 -> Vai para a Direita. Q4.
  - Se for ε ou M -> Vai para a Esquerda. Q5.
- Q5
  - Se for 1 -> Assina M. Vai para a Esquerda. Q3.
  - Se for M -> Vai para a Esquerda. Aceita.

Halting Problem

Máquina de Turing, para alguns casos, processava indefinidamente (sem nunca parar).

Não existe algum procedimento automático.

$P = \{ x | x \in N\ x \text{ é } primo \}$

Reformulando sob a forma de um problema de decisão:

$N = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, \ldots\}\ \sigma = \{ 0, \ldots, 9 \}\ N\ contém\ \sigma^*$

Caso do dia 18/02/2019, continuação:

$\forall w \in L,\ M\ aceita\ (w)$

$\forall w' \ni L,\ M\ aceita\ (w)$

- Q2
  - Se for 1 -> Vai para a Direita. Rejeita.
- Q5
  - Se for 0 -> Vai para a Direita. Rejeita.

Máquina de Turing Determinísticas

$$w \rightarrow M \rightarrow \{Aceita,\ w \in L\ Rejeita,\ w \ni L\ ,\ \forall w \in \sigma^*$$

Compromisso em Reconhecer ambos: Linguagem e $\neg$Linguagem

IMPORTANTE: Toda Máquina de Turing Determinística é uma Máquina de Turing Não-Determinística

Máquina de Turing Não-Determinísticas

$$w \rightarrow M \rightarrow \{Aceita,\ w \in L\ Rejeita\ ou\ Loop,\ w \ni L\ ,\ \forall w \in \sigma^*$$

Compromisso em reconhecer a Linguagem

Turing havia conhecimento matemático de conceitos que, hoje, capturam os conceitos básicos da informática.

Função

$$A = \{1, 2, 3\}\ B = \{a, b, c\}\ f : A \rightarrow B$$

Hipótese de Turing – Church

"...A tese de Turing- Church diz respeito à noção de um método mecânico e efetivo no contexto da Lógica e Matemática.

Um método, ou procedimento, M, que é empregado para se obter um resultado desejado é denominado “efetivo” ou mecânico.

Assim M:

- Deve ser descrito em um número finito de instruções exatas. (Cada instrução sendo expressa por um número finito de símbolos);
- Se M não apresentar erros, deve produzir o resultado desejado em um número finito de etapas;

- M pode ser realizado por um ser humano sem o auxílio de uma máquina, ou seja, empregando apenas “lápis e papel”;
- O método M não deve exigir nenhuma inferência e nem mesmo inventividade do ser humano que o executa." (Zalta, E. N.)

Exercício Resolvido: Relativamente à Hipótese de Church, responda:

a) Quais as implicações da Hipótese de Church?

Pela Hipótese de Church, as máquinas de Turing são versões formais dos algoritmos. Adotar tal modelo de algoritmo implica na possibilidade de formalmente se provar que certos problemas não podem ser resolvidos por nenhum algoritmo.

b) Por que a Hipótese de Church é assim denominada, em vez de Teorema de Church?

A Hipótese de Church não é um resultado formalmente provado.

Conjuntos Contáveis

Conjuntos Contáveis são aqueles em que é possível relacionar cada um de seus elementos com um número natural.

- Conjuntos finitos são contáveis.
- Alguns conjuntos infinitos são contáveis, por exemplo:
    - Descrevendo uma relação por 2n (função):

$T = \{0,\ 2,\ 4,\ 6,\ 8,\ 10,\ 12,\ 14,\ 16,\ 18,\ \ldots\}$ $N = \{0,\ 1,\ 2,\ 3,\ 4,\ 5,\ 6,\ 7,\ 8,\ 9,\ \ldots\}$

- Conjuntos Racionais Q:

$Q = \{\frac{1}{1}, \frac{1}{2}, \frac{1}{3}, \frac{1}{4}, \ldots, \frac{2}{1}, \frac{2}{2}, \frac{2}{3}, \ldots, \frac{N}{M}, \ldots\}$

1	$\frac{1}{1}$	$\frac{1}{2}$	$\frac{1}{3}$	$\frac{1}{4}$	$\frac{1}{5}$	$\frac{1}{6}$	$\frac{1}{7}$	.
2	$\frac{2}{1}$	$\frac{2}{2}$	$\frac{2}{3}$	$\frac{2}{4}$	$\frac{2}{5}$	$\frac{2}{6}$	$\frac{2}{7}$	.
3	$\frac{3}{1}$	$\frac{3}{2}$	$\frac{3}{3}$	$\frac{3}{4}$	$\frac{3}{5}$	$\frac{3}{6}$	$\frac{3}{7}$	.
4	$\frac{4}{1}$	$\frac{4}{2}$	$\frac{4}{3}$	$\frac{4}{4}$	$\frac{4}{5}$	$\frac{4}{6}$	$\frac{4}{7}$	.
.	.	.	.	.	.	.	.	

Portanto Q também é contável.

- Alguns conjuntos infinitos são incontáveis:
    - $C = \{\ x\ |x \in R\ e\ 0 \leq x \leq 1\ \} = \{\ 0,\ \ldots,\ 0.1,\ \ldots,\ 0.2,\ \ldots,\ ?,\ 1\ \}$

Existem infinitos elementos entre cada elemento de C, portanto ele é incontável.

Alfabeto

$\sigma = \{a, b\}$ $\sigma^0 = \{\varepsilon\}$ $\sigma^1 = \{a, b\}$ $\sigma^2 = \{aa, ab, ba, bb\}$ $\sigma^3 = \{aaa, aab, \ldots, bbb\}$ $\sigma^* = \sigma^0 \cup \sigma^1 \cup \sigma^2 \cup \sigma^3 \cup \ldots = \{\varepsilon,\ a, b, aa, ab,$

Linguagem

$L_1 = \{a,\ aa,\ aaa,\ aaaa,\ \ldots\}$ é contável $L_2 = \{ab,\ abb,\ abbb,\ abbbb,\ \ldots\}$ é contável $L_0 = \{a\}$ é contável

Conjunto das Partes

O conjunto de todos os subconjuntos

O conjunto $A = \{a, b\}$ tem como Conjunto das Partes $P = \{\ \varnothing,\ \{a\},\ \{b\},\ \{a, b\}\ \}$.

Como o P é um Conjunto das Partes de A, que é um conjunto **finito**, então P também é **finito**.

Se fosse o fechamento $\sigma^* = \{\varepsilon,\ a, b, aa, ab, ba, bb, \ldots\}$, aí sim seria **infinito**, pois seu Conjunto das Partes representa incontáveis conjuntos dos subconjuntos, onde:

$P = \{\varnothing,\ \{\varepsilon\},\ \{a\},\ \{b\},\ \{aa\}, \{ab\}, \ldots, \{\varepsilon, a\}, \{\varepsilon, b\}, \{\varepsilon, aa\}, \ldots.\ \{\varepsilon, a, a\}, \{\varepsilon, a, b\},\ \{\varepsilon, a, aa\}, \{\varepsilon, a, ab\}, \ldots\}$

Levando em conta o Fechamento σ^* de um alfabeto A, **todos os SUBCONJUNTOS encontrados em seu Conjunto de Partes são, portanto, uma linguagem**:

- $\{\varepsilon\}$ é uma linguagem;
- $\{a\}$ é uma linguagem;
- $\{b\}$ é uma linguagem;
- $\{aa\}$ é uma linguagem;
- $\{\varepsilon, a, a\}$ é uma linguagem; etc.

O conjunto de todas as linguagens dadas sob um alfabeto não é contável.

Problemas Não Solucionáveis sobre as Máquinas de Turing e sobre as Gramáticas.

Problema Solucionável (ou decidível)

Um problema é dito Solucionável ou Totalmente Solucionável se existe um algoritmo que solucione o problema tal que sempre pára para qualquer entrada, com uma resposta afirmativa (aceita) ou negativa (rejeita).

Problema Não-Solucionável (Não decidível)

Um problema é dito Não-Solucionável se não existe um algoritmo que solucione o problema tal que sempre para, qualquer que seja a entrada.

Problema Parcialmente Solucionável ou Computável

Um problema é dito Parcialmente Solucionável ou Computável Se existe um algoritmo que solucione o problema tal que pare quando a resposta é afirmativa (aceita). Entretanto quando a resposta esperada for negativa, o algoritmo pode parar (rejeita) ou permanecer processando indefinidamente.

Problema Completamente Insolúvel

Um problema é dito Completamente Insolúvel ou Não-Computável se não existe um algoritmo que solucione o problema tal que pare quando a resposta é afirmativa.

É importante observar que alguns problemas não-solucionáveis são parcialmente solucionáveis.

Ainda, existem problemas não-solucionáveis que possuem solução parcial.

A Classe dos Problemas Solucionáveis é equivalente à Classe das Linguagens Recursivas.

A Classe dos Problemas Parcialmente Solucionáveis é equivalente à Classes das Linguagens Recursivamente Enumeráveis.

Problemas Não solucionáveis sobre Máquinas de Turing

Provam-se insolúveis os seguintes problemas sobre gramáticas:

Dada uma máquina de Turing M e uma palavra w, ambas arbitrárias, M para com uma entrada w?

Dada uma máquina de Turing M, M para com a fita de entrada vazia?

Dada uma máquina de Turing M arbitrária, existe uma palavra que a faça parar?

Dada uma máquina de Turing M arbitrária, M para com qualquer entrada possível?

Dadas duas máquinas de Turing M1 e M2 arbitrárias elas param com as mesmas entrada".

Dada M arbitrária, a Linguagem por ela aceita é regular? É livre de contexto? É recursiva?

Exercício Resolvido : É um exemplo de problema não solucionável:

a) Não existem problemas não solucionáveis.

b) Minimização de um autômato finito.

c) Detector universal de loops.

d) Reconhecimento de linguagens recursivas.

e) Tratamento de não determinismos em linguagens regulares.

Resposta: a alternativa correta é a c). O detector universal de loops, diz respeito ao Problema da Parada.