Modelos Abstratos (Matemáticos) de maquinas

-funções recursivas
-maquinas de estados finitos

Funções Recursivas

-Definições Básicas
-Funções Totais e Parciais
-Composição de Funções de mais de um argumento

-Funções Iniciais

Definição: Seja f(x1,x2,x3,x4......Xn) uma função de n argumentos números naturais . Se f for definida para todos os números naturais como f:N^n N então f é chamada uma função TOTAL.

Por outro lado, se a função f for f:D -> N onde D(esta contido/ igual)N^n, então f é chamada uma função PARCIAL.

Exemplos: f(x,y) = x + y é definida para todo x,y E N e portanto é uma função total.

F(x,y) = x-y é definida apenas por aqueles x e y E N que satisfaçam x>= y e portanto é uma função parcial.

Composição de funções de mais de um argumento

- A composição de função e é utilizada para gerar outras funções.
- A idéia de composição de funções pode ser extendida para funções de mais de um argumento.

Exemplo: Sejam f1(x,y) , f2(x,y) e g(x,y) quaisquer três funções. A composição de g com f1 e f2 é uma função H dada por :

H(x,y) = g(f1(x,y),f2(x,y))

Generalizando:

Sejam f1,f2...fn funções de m argumentos e seja g uma função de n argumentos. Então , a composição de g com f1...f2......fn produz uma função H dada por:

H(x1,x2......xn) = g(f1(x1,x2....xm), f2(x1,x2,x3....xm),.....fn(x1,x2,x3....xm))

Exemplo: Sejam f1(x,y) = x+y, f2(x,y) = xy+y^2 e g(x,y) = xy

H(x,y) = g(f1(x,y),f2(x,y)) = g((x+y),(xy+y^2)) = (x+y)(xy+y^2) = x^2y+xy^2+xy^2 + y^3 <u>= x^2y+2xy^2+y^3.</u>

Funções Iniciais - São funções que são utilizadas para definir outras funções por recursão.

1. FUNCAO ZERO : z: N → N z(x) = 0
2. FUNCAO SUCESSOR: s: N → N s(x) = x+1
3. FUNCAO PROJECAO OU FUNCAO IDENTIDADE GENERALIZADA
   Ui:N^n → N Ui^n(x1,x2,x3....xn) = xi;

Exemplo: i=1 Ui^1(x) = x   i=2 Ui^2(x,y) = y   i=2 Ui^4(x,y,z,w) = y

Dada uma função f(x1,x2,x3,.....xn), muitas vezes é conveniente considerar n-1 destas variáveis como fixas e varis apenas o ultimo argumento sobre o domínio dos números naturais.

Apesar de parecer extremamente trabalhosa num processo de calculo manual essa técnica pode ser bastante interessante em um processo de computação automático.

Assume-se que possuímos um mecanismo para avaliar o valor da função quando o ultimo argumento for zero, bem como o valor da função quando o ultimo argumento valer n+1 através do valor da função quando o ultimo valor vale n.

Ou seja : uma função pode ser definida pelo processo de recursão, definindo-se duas outras funções conhecidas.

F(x1,x2,.....,xn)

1. A função base:

F(x1,x2,....xn-1,0) = g(x1,x2,...xn-1)

2. A função passo-indutivo

F(x1,x2.....xn-1,<u>xn</u>+1) = H(x1,x2,....,xn,f(x1,x2....xn))

Exemplo : Seja f(x1,x2) = x1+x2

Assumimos que f(x1,0) = x1

Também sabemos que f(x1,x2+1) = 1 + f(x1,x2)
Assim f(2,3) = 1 + f(2,2) = 1+[1+f(2,1)] = 1+[1+[1+F(2,0)]] = 1+[1+[1+2] = 1+1+3 =
= 1+4 = 5
Ou seja,

1. Função base :

F(x1,0) = i=1 Ui^i(x)

2.Função Passo-Indutivo

F(x1,x2+1) = S(U_3^3(x1,x2,f(x1,x2)))

Definição: Uma função f é chamada primitiva recursiva se e somente se ela puder ser obtida de funções iniciais através de um numero finito de operações de de composição e recursão.

Definição : Uma função f é chamada recursiva se e somente se ela puder ser obtida de outras funções recursivas ou primitivas recursivas através de um numero finito de operações de composição e recursão.

Exemplo : Mostre que a função m: N² -> N definida como m(x,y) = xy é recursiva.

1. Função base : m(x,0) : Z(x)
2. Função passo-indutivo

$m(x,y+1) = f(U_1^3(x,y,m(x,y))), U_3^3(x,y,m(x,y))$

Onde f(x,y) é a função soma, que já mostramos ser primitiva recursiva.

MODELOS ABSTRATOS DE FUNÇOES RECURSIVAS

Funções Iniciais
- Função Zero
- Função Sucessor
- Função Projeção ou Identidade Generalizada

Funções Recursivas
Função base:
F(x1, x2,...,xN-1,0) = g(x1, x2,...,xN-1)
Onde o valor de g(x1, x2,...,xN-1) é conhecido ou pode ser calculado.

Função Passo Indutivo:
f(x1,x2,...,xN +1) = h(x1,x2,...,xN, f(x1,x2,...,xN))
onde o valor de h(x1,x2,...,xN, f(x1,x2,...,xN)) é conhecido ou pode ser obtido através de um numero finito de operações de composição e recursão de funções.

Exemplos: $f:N^2 \to N$ f(x,y) x+y

Função Base:
$F(x,0) = U^1{}_1(x)$

Função Passo Indutivo:
$F(x,y+1) = S(U^3{}_3(x,y,f(x,y))$

Obs.: Note que x+(y+1) = (x+y)+1

Fat: N-> N fat (x)=x!

Note que: x!=x.(x-1)!

Função Base: fat(0)=1

Função Passo-Indutiva

$Fat(x+1)=m(S(U_1^2(x,fat(x))),U_2^2(x,fat(x)))$

Outras funções recursivas:

1. Função sinal: Sg: N->N Sg(x)=1 se x>0 ou Sg(x)=0 se x=0
   F.B. Sg(0)=0
   F.P.I. $Sg(x+1)= S(Z(U_1^2(x,sg(x))))$

2. Funções Testa-Zero: ~Sg: N->N ~Sg(x)=0 se x>0 ou ~Sg(x)=1 se x=0
   F.B. ~Sg(0)=1
   F.P.I. $\sim Sg(x+1)=Z(U_1^2(x,\sim Sg(x)))$

3. Função Antecessor: A: N->N A(x)=0 se x=0 ou A(x)=x-1 se x>0
   F.B. A(0)=0
   F.P.I. $A(x+1)=U_1^2(x,A(x))$

4. Função Subtração Própria: -.:N^2->N

   -. (x,y) = 0 se y>x ou -. (x,y) = x-y se y<=x

   F.B. -. $(x,0)=U^1_1(x)$
   F.P.I. Note que x-(y+1) = A(x-.y)
   P. Ex 5-.3 = A(5-.2) = 2

   -.(x,y+1) = $A(U^3_3(x,y,-.(x,y))$

5. Função Mínimo: min: N^2 -> N min(x,y)=x se x<=y ou min(x,y)=y se x>y
   F.B. min(x,o)=Z(x)
   F.P.I. min(x,y+1) = $-.(U^3_1(x,y,\min(x,y)),A(-.(U^3_1(x,y,\min(x,y)),U^3_2(x,y,\min(x,y))))$

   min(x,y+1) = -.(x,A(-.(x,y)))

6. Função Máximo: max:N^2 -> N max(x,y)=x se x>=y ou max(x,y)=y se x<y
   F.B. max(x,0) = $U^1_1(x)$
   F.P.I.
   max(x,y+1) = $f(S(U^3_2(x,y,\max(x,y)),A(-.(U^3_1(x,y,\max(x,y),U^2_2(x,y,\max(x,y))))$

   max(x,y+1) = (y+1)+A(-.(x,y))

Função Quadrática q:N -> N $q(x)=x^2$
   F.B. q(0) = 0
   F.P.I. q(x+1) = $m(S(U^2_1(x,q(x))),S(U^2_1(x,q(x))))$

   Obs.: $q(x+1) = (x+1)^2 = (x+1)*(x+1) = m(x+1, x+1)$

7. Função Paridade: p:N -> N p(x)=1 se x é par ou p(x)=0 se x não é par
   F.B. p(0)=1
   F.P.I. p(x+1) = $\sim sg(U^2_2(x,p(x))$

8. Função resto da divisão por 4: mod_4: N -> N mod_4 = x mod 4
   F.B. $mod_4(0) = 0$
   F.P.I. $mod_4(x+1) =$

Linguagens e Gramáticas

Para definir uma linguagem:

- Alfabeto
- Palavras ou cadeias de caracteres

A especificação da construção apropriada de sentenças é chamada sintaxe da linguagem.

Estudaremos a sintaxe de uma classe de linguagens chamadas gramática com estrutura de frase.

Modelos abstratos de máquina
Linguagens

Gramática com estrutura de frase

- Definição 1: Um vocabulário(ou alfabeto) V(vezão) é um conjunto finito e não-vazio de elementos chamados símbolos. Exemplo: letras e dígitos

Uma palavra(ou sentença) sobre V é uma cadeia de tamanho finito sobre V. A cadeia vazia ou nula, denotado por λ, é a cadeia que não contém símbolos.

O conjunto de todas as palavras sobre V é denotado por V*.

Exemplo: Se V = {a,b} então:

V* = {λ, a, b, aa, ab, bb, aaa, …}

Uma linguagem sobre V é um subconjunto de V*.

Linguagens podem ser especificadas de várias formas:

a) Listando todas as palavras da linguagem.

b) Fornecendo um certo critério que deve ser satisfeito pelas palavras que pertencem a linguagem.

c) Através do uso de uma gramática.

<u>Gramáticas</u>

Uma gramática provê um conjunto de vários tipos de símbolos e um conjunto de regras para produção de palavras.

Uma <u>gramática</u> possui um vocabulário V, que é o conjunto de símbolos utilizados para derivar membros da linguagem.

Alguns elementos do vocabulário não podem ser substituídos por outros símbolos. Eles são chamados de terminais.

Os outros símbolos do vocabulário, que podem ser substituídos por outros símbolos, são chamados de não-terminais. Os conjuntos de terminais e não-terminais são denotados por T e N respectivamente.

Existe um membro especial do vocabulário chamado <u>símbolo inicial</u>, denotado por S.

As regras que especificam quando podemos substituir uma cadeia de V* por outra String são chamadas de produções da gramática. Denotamos por Z0(zê-zero) → Z1(zê-um) a produção que especifica que Z0 pode ser substituído por Z1 em uma cadeia.

Definição 2: Uma gramática com estrutura de frase G = (V, T, S, P) consiste de um vocabulário V, um subconjunto T de V de elementos terminais. Um símbolo inicial S de V e de um conjunto de produções P. Toda produção em P deve conter somente símbolos não-terminais do seu lado esquerdo.

Exemplos:

Seja G = (V,T,S,P) onde:

V = {a, b, A, B, S}

T = {a, b}

S é o símbolo inicial

P = { S → Aba,
A → BB,
B → ab,
AB → b }

L(G) = { ba, abababa}

Definição 3: Seja G = (V, T, S, P). Sejam w0(w-zero) = Iz0R(ou seja, a concatenação de I, z0 e R) e w1(w-um) = Iz1R cadeias sobre V. Se w1 é diretamente derivável de w0 e escrevemos w0 => w1. Se w0, w1, …, wn são cadeias sobre V tal que w0 => w1 => … => wn, então dizemos que wn é derivável de w0 e escrevemos w0 => wn. A sequência de passos utilizados para obter wn a partir de w0 é chamada de derivação.

Exemplo: Seja G a gramática com V = {S, A, a, b} , T = {a, b}, S = S e P = {S → aA,
S → b,
A → aa}

Qual a linguagem L(G) desta gramática?!

L(G) = {aaa, b}

Definição 4: Seja G = (V, T, S, P). A linguagem gerada por G (ou a linguagem de G), denotado por L(G) é o conjunto de todas é conjunto de todas as cadeias de terminais que são deriváveis a partir do estado inicial S, ou seja, L(G) = {w E(pertence) T* | S => w}

- Exemplos de concatenação; Seja w1 = a e w2 = bc.
  - $w1^5$ = aaaaa
  - $w2^4$ = bcbcbcbc

- Exemplo: Seja G a gramática com V = { S, 0, 1}, T = {0, 1}

  S = S e P = {S → 11S, S → 0}
  L(G) = ?

  L(G) = {0, 110, 11110, 1111110, 111111110, …}

  L(G) = {1^N0 | n é par}

LINGUAGENS E GRAMÁTICA

Exemplo: Ache a gramática com estrutura de frase que gere o conjunto $\{o^{m}{}_{1}{}^{n}\}$ m e n pertencentes a Z^{+}.

V = {
T = {
P = { S → 0S, A → 1, S → 1ª, S → λ, S → 1, A → 1A}
S = S

P = { S□0S, S□S1, S□ λ}

<u>Tipos de Gramática com estrutura de frase</u>

-Uma gramatica do tipo 0(zero) não tem restrições suas produções

-Uma gramática do tipo 1 pode possuir produções de forma w_1 □ w_2 onde o tamanho de w_2 é maior que o tamanho de w_1 ou da forma w_1 □ λ

-Uma gramatica do tipo 2 pode possuir produções da forma w_1 □ w_2 onde w_1 é único símbolo e é não-terminal

-Uma gramática do tipo 3 pode possuir produções apenas da forma w_1□w_2 com w_1 =A e tanto w_2 =aB ou w_2 =a onde A e B são símbolos não-terminais e a é um símbolo terminal, ou ainda com w_1 = S e w_2 = λ.

Gramaticas do tipo 3 são também chamados de gramática regular. Uma linguagem gerada por uma gramática regular é chamada linguagem regular

Exemplo : $o^{m}{}_{1}{}^{n}$ é uma linguagem tipo 2. Porque?

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

Em estudos anteriores, consideramos o conjunto V* consistindo de todas cadeias(Strings) de elementos de um conjunto(alfabeto) V.

Se pensarmos em V como um conjunto de “ palavras”, então V* pode ser considerado como o conjunto de todas as possíveis “sentenças” formadas a partir de palavras de V. Claro que tais sentenças não necessitam ter um significado.

Devemos pensar em uma linguagem como uma especificação completa, pelo menos em principio, de 3 elementos.

1. Deve existir um conjunto de palavras que fazem parte da linguagem

2. Um subconjunto de V* deve ser designado como o conjunto de ‘’sentenças corretamente construídas’’ da linguagem.

3. Deve ser determinado quais das sentenças corretamente construídas tem significado e qual seu significado.

<u>Comprimento de cadeias</u>

Seja W_n uma palavra em {a,b}*

Exemplos: w_1 = aab w_2 = abbba w_3 = ababababaaa

O comprimento de W é representado por |w|, assim: | w_1|=3 | w_2|=5 | w_3 |= ?

<u>Prefixo, sufixo subpalavra e expressão regular</u>

Se w uma palavra de {a,b}* e w= aabab

Conjunto dos prefixos de w: P={ a, aa, aab, aaba,aabab}

Conjunto dos sufixo de w: S={b, ab,bab,abab,aabab}

Conjunto das subpalavras de w: D = P U(união) S

Expressão regular $a^*bc^+b^2$

*Qualquer quantidade de vogais(inclusive zero vezes).

+ uma ou mais vezes

X onde x pertence Z^+, x vezes

Exemplo: Seja V o conjunto dos inteiro , dos símbolos +,-,x e ÷ e os parênteses().

Qual é uma das gramáticas que produzem as sentenças:

--   ((3-2)+(4x7)) ÷9

--   (7-(8-(9-10)))

Exercício: Para a gramática G=(V,T,S,P) com: V={E,i(maiúsculo),a,b,+,x} T={a,b, +,x}

S=E

P={E□i(maiúsculo)|E+E|ExE

i(maiúsculo)□a|b|i(maiúsculo)a|i(maiúsculo)b}

Encontre a arvore de derivação da palavra a+bxa

Autômatos Finitos – maquinas de estados finitos

Uma maquina de estados finitos ou automato finito é uma modelagem de um comportamento, composto por estados , transições e acoes.

Um estado armazena informações sobre o passado, isto é, ele reflete as mudanças desde a entrada em um estado, no inicio do sistema, até o momento presente.

Uma transição indica uma mudança de estado e é descrita por uma condição que precisa ser realizada para que a transição ocorra.

Uma ação é a descrição de uma atividade que deve ser realizada em um determinado momento.

Maquinas de estados finitos poder ser representadas por meio de um diagrama de estados (ou diagrama de transição de estados).

Diversas tabelas de transição de estados são usadas. Através do uso das tabelas podemos representar uma de maquina finita de estados que contenha informações completas sobre as ações.

Um automato é representado pela quíntupla (Q,V,DELTA,S,F) onde:

Q: é um conjunto finito de estados.

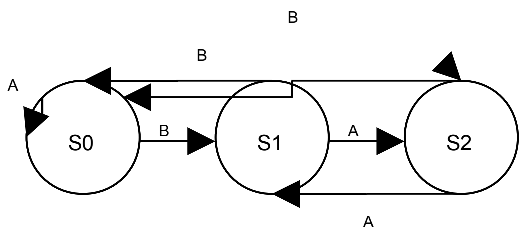
V: é o vocabulário (alfabeto) de caracteres pertencentes a linguagem.

DELTA: é a função de transição onde, dado um estado atual q1EQ, existe um simbolo v1EV que leva ao estado q2(q1Xv1 → q2).

S: É O ESTADO inicial.

F: é o conjunto de estados finais, estados onde o autômato reconhece determinada sequência de caracteres.

	a	b
S0	S0	S1
S1	S2	S0
S2	S1	S0



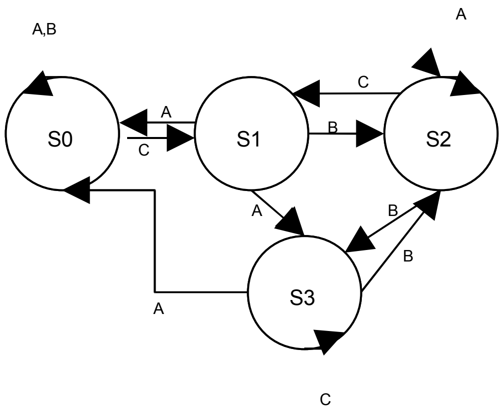
Exemplo: Dada uma maquina de estados finitos M = (Q,V,DELTA, S,F) onde:

Q: {S0,S1,S2,S3}
V: {a,b,c}
S = S0

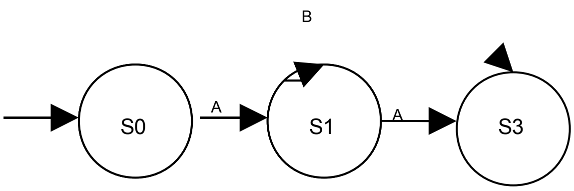
DELTA	a	b	c
S0	S0	S0	S1
S1	S0	S2	S3
S2	S2	S3	S1
S3	S0	S2	S3

F : {S1,S3}

(a|b)*c



Exercício: Qual é a linguagem reconhecida pelo automato abaixo?



Exercício: Seja M = (Q,V,DELTA,S,F) onde:
Q:{S0.S1,S2}
V = {0,1}

DELTA	0	1
S0	S1	S0
S1	S2	S0
S2	S2	S0

S = S0
F = {S2}

Desenhe o diagrama de estados de M e diga quais cadeias de caracteres M reconhece.

Exercício : Seja M um automato finito que reconheça números pares em binário Desenhe o diagrama de estados de M. De a formalização de M (Q,V,DELTA,S,F).

Exercício : Desenhe os diagramas de estados dos automatos que reconhecem as linguagens a seguir:

a) L1 = {wE{0,1}* | w contem pelo menos três 1's}
b) L2 = {wE{0,1}* | w contem um numero impar de símbolos}.

Modelos Abstratos de Maquinas

Maquinas de Estados Finitos

MAQUINA DE MOORE

Representado pela quíntupla:

M = (Q,V,DELTA,S,F) onde:

Q = um conjunto finito de estados
V = é o vocabulário (alfabeto) de caracteres pertencentes a linguagem
DELTA = função de transição de estados.
S = Q x V → Q
S = é o estado inicial
F = conjunto de estados finais, estados nos quais o autônomo reconhece determinada sequência de caracteres.

MAQUINA DE MEALEY

Representado pelo quíntuplo:

N = (I,Q,O,DELTA,LAMBÍDA) onde:

I = alfabeto de entrada
Q =conjunto finito de estados
O = alfabeto de símbolos de saída
DELTA = função de transição de estados
DELTA = Q x I → Q
LAMBÍDA = função de saída
LAMBÍDA = Q x I → O

Exemplo: Máquina de refrigerante
I={0,5 , 1,0}
O={“-1,5”, “-1”, “-0,5”, “R,-2,0”, “R,+0,5”}
Q={S0, S1, S2, S3}
DELTA={(S0, 0,5, S1), (S0, 1, S2), (S1, 0,5, S2), (S1, 1, S3), (S2, 0,5, S3), (S2, 1, S0), (S3, 0,5, S0), (S3, 1, S0)}
LAMBÍDA={(S0, 0,5, -1,5), (S0, 1, -1), (S1, 0,5, -1), (S1, 1, -0,5), (S2, 0,5, -0,5), (S2, 1, R-2,0), (S3, 0,5, R-2,0), (S3, 1, R+0,5)}

<Diagrama de estados formado a partir de DELTA e LAMBÍDA>