Combinação de modelos

Marco A L Barbosa
malbarbo.pro.br

Departamento de Informática
Universidade Estadual de Maringá

Introdução

Introdução

- Qual modelo utilizar quando a função consome dois ou mais tipos de dados?
  - Se apenas um dado é definido por mais que uma cláusula (como por exemplo, uma lista), utilizamos o modelo correspondente
  - Se mais que dois dados de entrada são definidos por mais que uma cláusula, devemos fazer uma combinação dos modelos

Exemplos

Exemplo 5.1

Dados duas listas `lsta` e `lstb`, defina uma função que verifique se `lsta` é prefixo de `lstb`, isto é `lstb` começa com `lsta`.

Exemplo 5.1

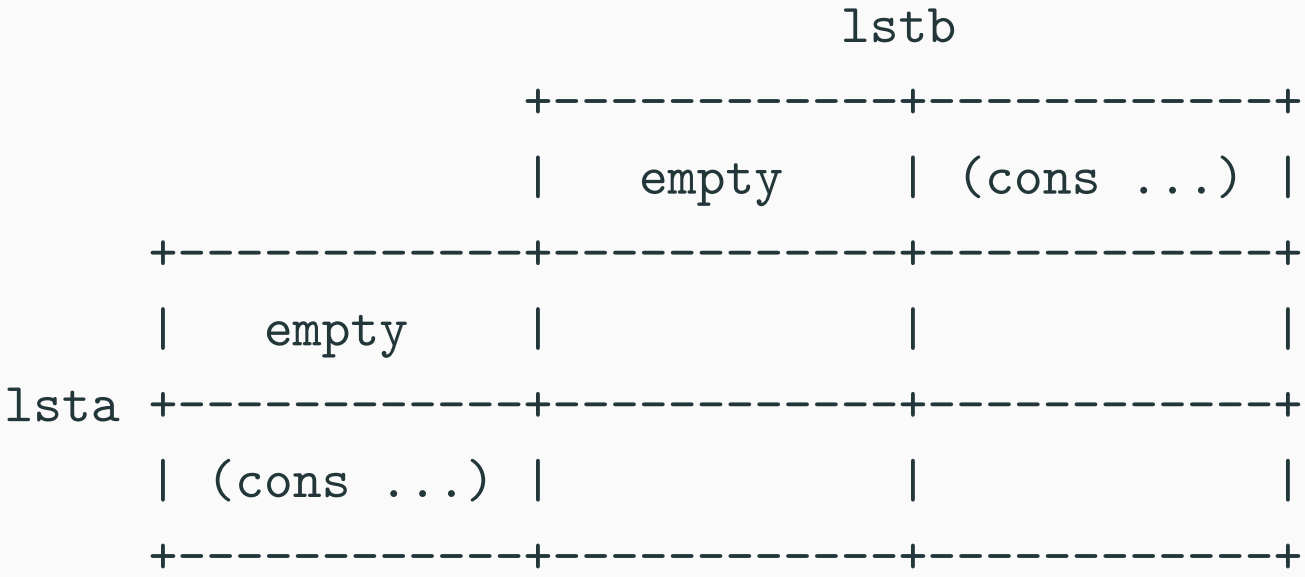
- Passo 1: Assinatura, propósito e cabeçalho

```
;; Lista Lista -> Boolean
;; Devolve #t se lsta é prefixo de lstb,
;; #f caso contrário.
(define (prefixo? lsta lstb) #f)
```

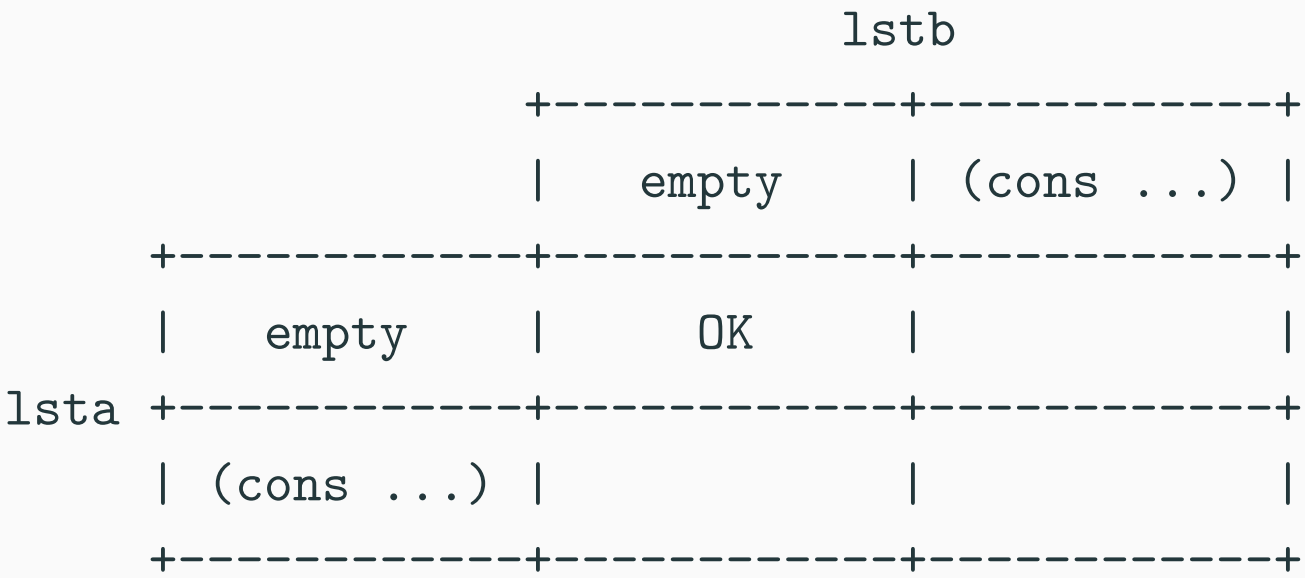
Exemplo 5.1

- Passo 2: Exemplos
  - Temos que ter pelo menos um exemplo para cada combinação das definições dos dados de entrada
  - `lsta` pode ser `empty` ou um `cons`
  - `lstb` pode ser `empty` ou um `cons`
  - Como garantir que não vamos esquecer nenhum caso? Fazendo uma tabela!

Exemplo 5.1

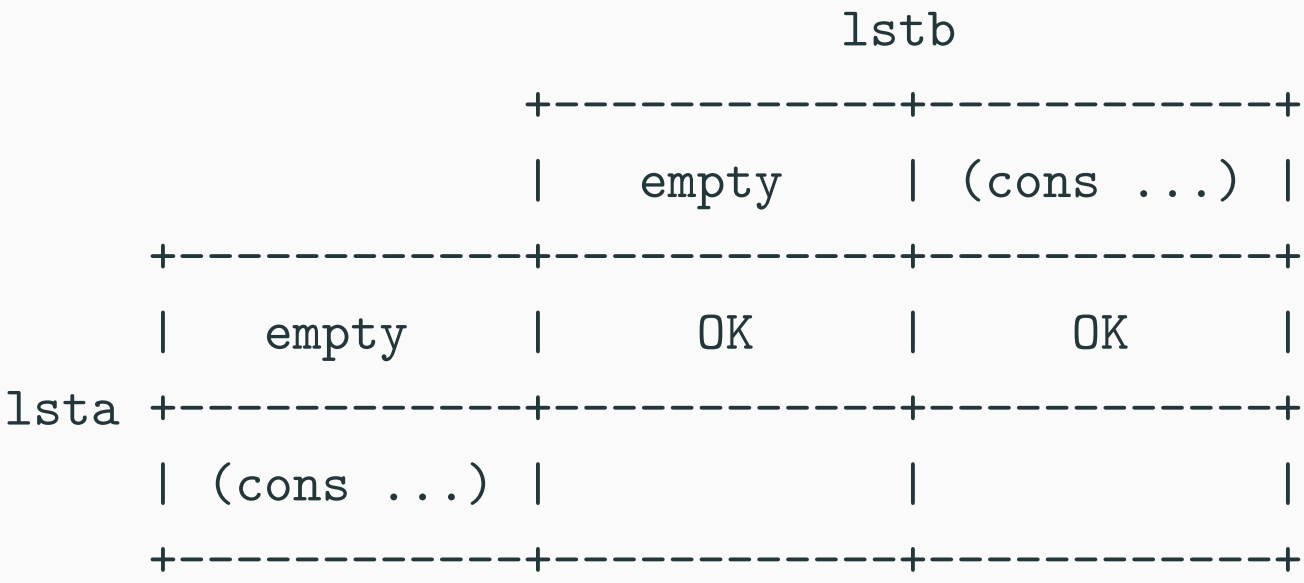


Exemplo 5.1



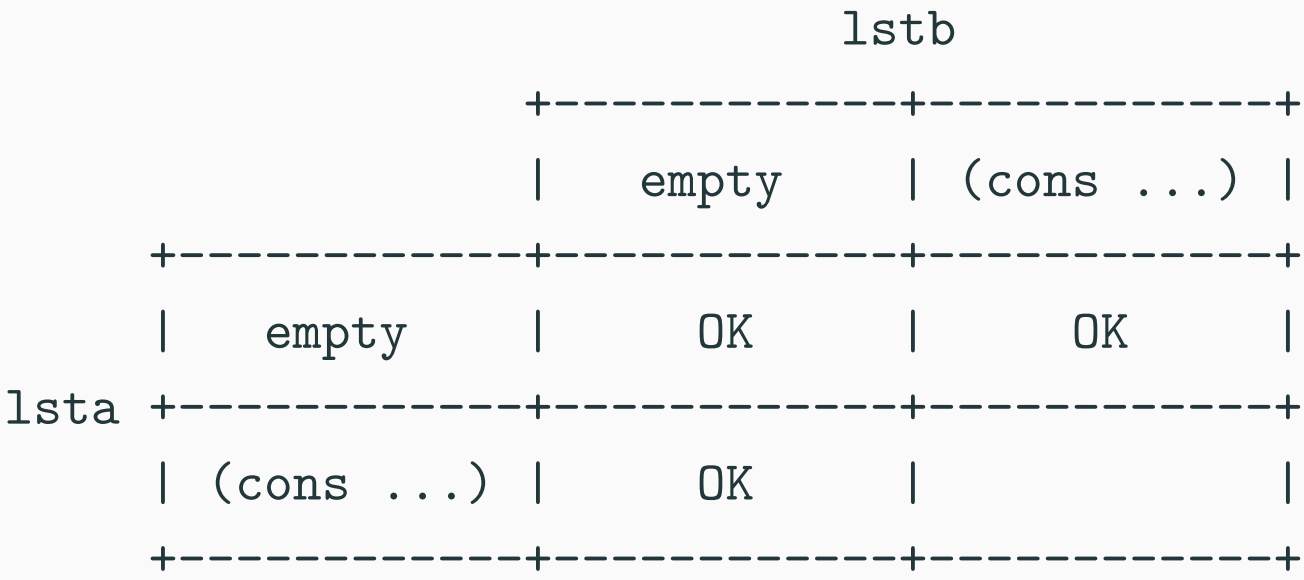

```
(check-equal? (prefixo? empty empty) #t)
```

Exemplo 5.1




```
(check-equal? (prefixo? empty empty) #t)
(check-equal? (prefixo? empty (list 3 2 1)) #t)
```

Exemplo 5.1




```
(check-equal? (prefixo? empty empty) #t)
(check-equal? (prefixo? empty (list 3 2 1)) #t)
(check-equal? (prefixo? (list 3 2 1) empty) #f)
```

Exemplo 5.1

lsta \ lstb	empty	(cons ...)
empty	OK	OK
(cons ...)	OK	OK

```
(check-equal? (prefixo? empty empty) #t)
(check-equal? (prefixo? empty (list 3 2 1)) #t)
(check-equal? (prefixo? (list 3 2 1) empty) #f)
(check-equal? (prefixo? (list 3 4) (list 3 4)) #t)
(check-equal? (prefixo? (list 3 4) (list 3 5)) #f)
(check-equal? (prefixo? (list 3 4) (list 3 4 6 8)) #t)
(check-equal? (prefixo? (list 3 5) (list 3 4 6 8)) #f)
(check-equal? (prefixo? (list 3 4 5) (list 3 4)) #f)
```

Exemplo 5.1

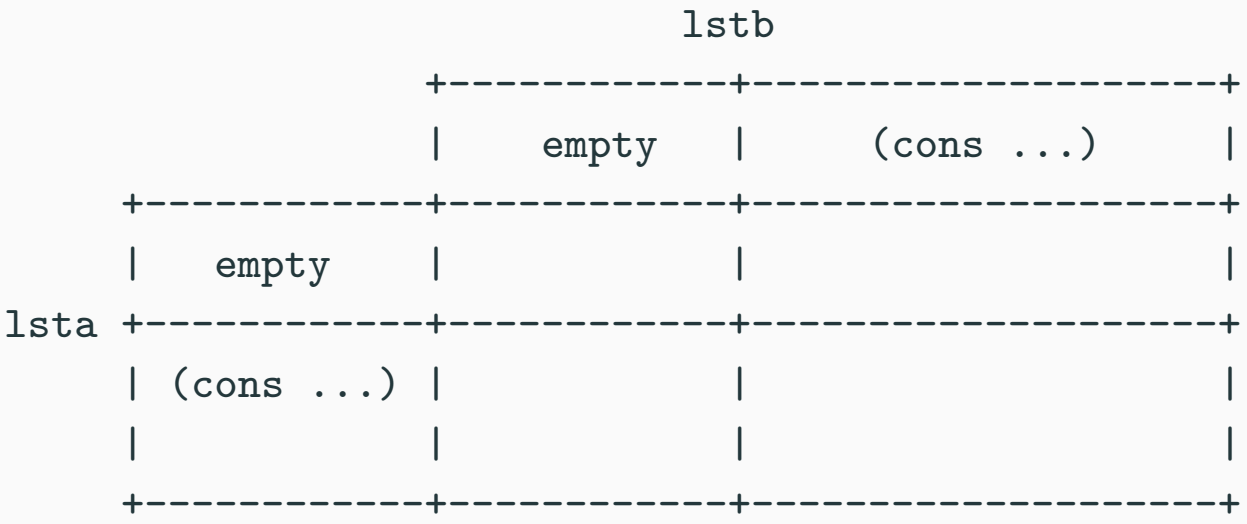
- Passo 3: modelo
  - Baseado na tabela, vamos criar um modelo

```
(define (prefixo? lsta lstb)
  (cond
    [(and (empty? lsta) (empty? lstb)) ...]
    [(and (empty? lsta) (cons? lstb)) ... lstb ...]
    [(and (cons? lsta) (empty? lstb)) ... lsba ...]
    [else ... lsta ... lstb ...]))
```

  - Este modelo é muito complicado...
  - Baseado nos exemplos, vamos preencher a tabela e derivar um modelo mais simples

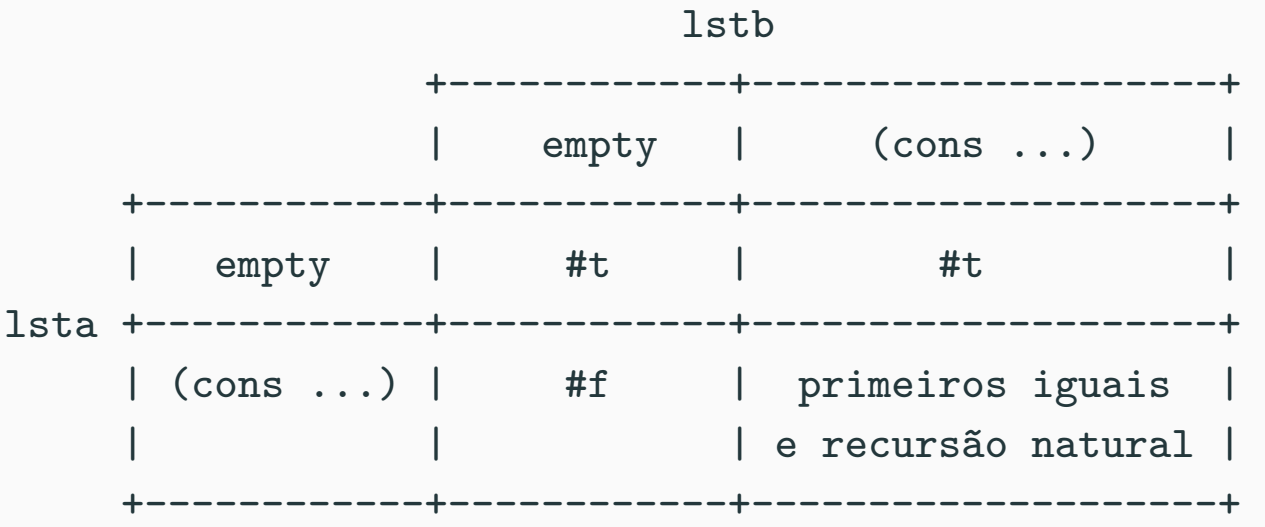
Exemplo 5.1

```
(check-equal? (prefixo? empty empty) #t)
(check-equal? (prefixo? empty (list 3 2 1)) #t)
(check-equal? (prefixo? (list 3 2 1) empty) #f)
(check-equal? (prefixo? (list 3 4) (list 3 4)) #t)
(check-equal? (prefixo? (list 3 4) (list 3 5)) #f)
(check-equal? (prefixo? (list 3 4) (list 3 4 6 8)) #t)
(check-equal? (prefixo? (list 3 5) (list 3 4 6 8)) #f)
(check-equal? (prefixo? (list 3 4 5) (list 3 4)) #f)
```

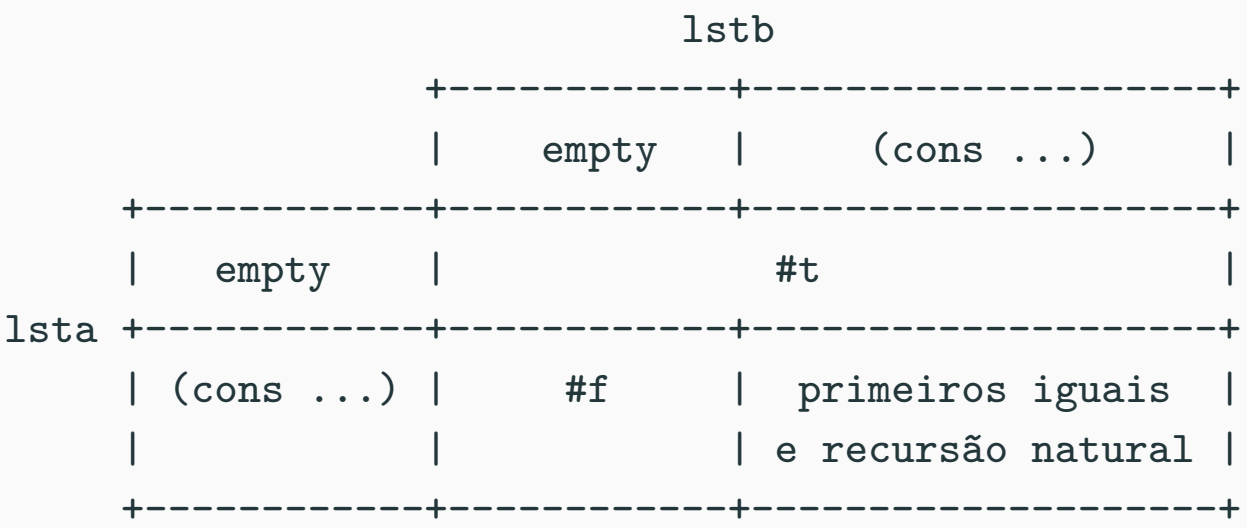


Exemplo 5.1

```
(check-equal? (prefixo? empty empty) #t)
(check-equal? (prefixo? empty (list 3 2 1)) #t)
(check-equal? (prefixo? (list 3 2 1) empty) #f)
(check-equal? (prefixo? (list 3 4) (list 3 4)) #t)
(check-equal? (prefixo? (list 3 4) (list 3 5)) #f)
(check-equal? (prefixo? (list 3 4) (list 3 4 6 8)) #t)
(check-equal? (prefixo? (list 3 5) (list 3 4 6 8)) #f)
(check-equal? (prefixo? (list 3 4 5) (list 3 4)) #f)
```



Exemplo 5.1 (simplificando...)




```
;; Modelo (observe que alguma parte do corpo já foi escrita)
(define (prefixo? lsta lstb)
  (cond
    [(empty? lsta) #t]    ;; os casos foram
    [(empty? lstb) #f]    ;; escolhidos por ordem
    [else ...             ;; de simplicidade
          (first lsta)
          (first lstb)
          (prefixo? (rest lsta) (rest lstb))]))
```

Exemplo 5.1

- Passo 4: Corpo

```
(define (prefixo? lsta lstb)
  (cond
    [(empty? lsta) #t]
    [(empty? lstb) #f]
    [else (and
            (equal? (first lsta) (first lstb))
            (prefixo? (rest lsta) (rest lstb)))]))
```

Exemplo 5.2

Defina uma função que encontre o k-ésimo elemento de uma lista.

Referências

Referências

- Vídeos 2 one-of