
INF05516 - Semântica formal N
Ciência da Computação - UFRGS
2006-2

Marcus Ritt
mrpritt@inf.ufrgs.br

09/08/2006

Introdução	2
Agenda	3
Nocões matemáticas	4
Nocões matemáticas (2).	5
Nocões matemáticas (3).	6
Sintaxe	7
IMP – uma linguagem pequena	8
IMP - vista geral.	9
IMP – Entidades sintáticas	10
Meta-variáveis	11
IMP – sintaxe abstrata.	12
Sintaxe abstrata e concreta.	13
IMP – observações	14
Semântica operacional	15
Idéia	16
IMP – categorias semânticas.	17
IMP – exemplo	18
Regras.	19
Expressões aritméticas	20
Instâncias.	21
Exemplo: expressões aritméticas	22
Equivalências	23
Expressões booleanas.	24
Expressões booleanas.	25
Exemplo: expressões booleanas	26
Comandos	27
Comandos	28
Exemplo: Comandos	29
Discussão e resumo	30

Agenda

Última aula:

- Introdução.

Hoje:

- Conceitos matemáticos básicos
- Repetir algumas noções das gramáticas
- Definir uma pequena linguagem imperativa
- Definir a sua semântica operacional
- Descobrir algumas características dela

v1915

Semântica formal N, aula 2 – 3 / 30

Nocões matemáticas

- Uma *relação* R entre dois conjuntos A, B é um conjunto $R \subseteq A \times B$
- Escrevemos também $a R b$ para $(a, b) \in R$.
- Uma relação é
 - ◆ *reflexivo* se $(a, a) \in R$
 - ◆ *transitivo* se $(a, b) \in R \wedge (b, c) \in R \rightarrow (a, c) \in R$
 - ◆ *simétrico* se $(a, b) \in R \rightarrow (b, a) \in R$
 - ◆ *anti-simétrico* $(a, b) \in R \wedge (b, a) \in R \rightarrow a = b$

Exemplo:

$$\leq \in \mathbb{N} \times \mathbb{N} = \{(1, 1), (1, 2), (1, 3), (2, 2), (1, 4), (2, 3), (1, 5), (2, 4), (3, 3), \dots\}$$

v1915

Semântica formal N, aula 2 – 4 / 30

Nocões matemáticas (2)

- Uma *função parcial* f do domínio D para o *imagem* I e uma relação entre D e I com *imagens únicas*:

$$(d, i_1) \in f \wedge (d, i_2) \in f \rightarrow i_1 = i_2$$

- O *domínio* (atual) é $\text{dom}(f) = \{d \mid (d, i) \in f\}$.
- A *imagem* (atual) é $\text{img}(f) = \{i \mid (d, i) \in f\}$.
- Uma função é total, se $\text{dom}(f) = D$.

Usamos a seguinte notação:

- Declaração: $f : D \rightarrow I$ se f e uma função do D para I
- Aplicação: $f(d) = i$ ou $d \mapsto_f i$, se $(d, i) \in f$

v1915

Semântica formal N, aula 2 – 5 / 30

Nocões matemáticas (3)

- Apresentamos uma função com domínio finito as vezes como

$$\{d_1 \mapsto i_1, d_2 \mapsto i_2, \dots\}$$

por $d_k \in \text{dom}(f)$ e $d_k \mapsto i_k$

- Escrevemos também

$$f[a \mapsto b](x) = \begin{cases} f(x) & \text{se } x \neq a \\ b & \text{se } x = a \end{cases}$$

para a função f modificada no ponto a .

- Escrevemos $[D \rightarrow I]$ para o conjunto das funções totais e $[D \rightharpoonup I]$ para o conjunto das funções parciais de D para I .

v1915

Semântica formal N, aula 2 – 6 / 30

IMP – uma linguagem pequena

- Vamos estudar uma linguagem imperativa simples – IMP
 - ◆ Sintaxa abstrata
 - ◆ Semântica operacional
 - ◆ Sistema de tipos
 - ◆ Semântica denotacional
 - ◆ Semântica axiomática
 - ◆ Relações entre várias semânticas

v1915

Semântica formal N, aula 2 – 8 / 30

IMP - vista geralEntrada: x, y

```
if y > 0 then
  while y < x
    x := x - y
else
  skip
```

Primeiro, precisamos definir a sintaxe.

v1915

Semântica formal N, aula 2 – 9 / 30

IMP – Entidades sintáticas

As entidades sintáticas são conjuntos das frases permitidos (literais, etc.).

- Int – números inteiros : `1,42,4711`
- Bool – booleanas: `true, false`
- Loc – locais: `x,y,a,b,c,...`
- Aexp – expressões aritméticas: `1+2×3, x+5`
- Bexp – expressões booleanas: `x=y^true, ¬false`
- Com – comandos:
`x := 5`
`if true then x:=1 else x:=0`

v1915

Semântica formal N, aula 2 – 10 / 30

Meta-variáveis

- Usamos *meta-variáveis* para elementos desses conjuntos:

$$\mathbf{n} \in \text{Int}, \mathbf{t} \in \text{Bool}, \mathbf{l} \in \text{Loc}, \mathbf{b} \in \text{Bexp}, \mathbf{a} \in \text{Exp}, \mathbf{c} \in \text{Com}$$

- Permitimos variações das meta-variáveis como

$$\mathbf{n}, \mathbf{n}', \mathbf{n}'', \mathbf{n}_0, \mathbf{n}_1, \mathbf{a}'$$

v1915

Semântica formal N, aula 2 – 11 / 30

IMP – sintaxe abstrata

- Expressões aritméticas

$$a ::= n \mid l \mid a + a' \mid a - a' \mid a \times a'$$

- Expressões booleanas

$$b ::= t \mid a = a' \mid a < a' \mid \neg b \mid b \vee b' \mid b \wedge b'$$

- Comandos

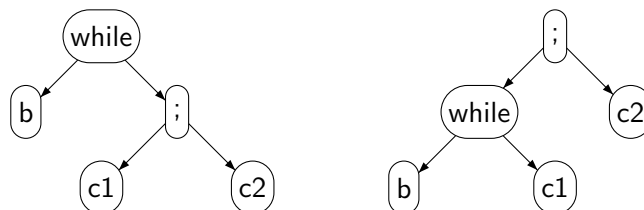
$$c ::= \text{skip} \mid l := a \mid c; c' \mid \text{if } b \text{ then } c \text{ else } c' \mid \text{while } b \text{ do } c$$

v1915

Semântica formal N, aula 2 – 12 / 30

Sintaxe abstrata e concreta

- A sintaxe abstrata denota *árvores de parse*
- Uma expressão linearizada é ambígua: `while b do c;c`



- A *sintaxe concreta* se cuida dessas ambigüedades: nos não; em caso de dúvidas usamos parênteses: `while b do (c;c)`

v1915

Semântica formal N, aula 2 – 13 / 30

IMP – observações

- As variáveis não são declarados
- As regras de tipos são embutidas na definição da sintaxe
- As expressões não tem efeitos colaterais
- Comandos causam os efeitos colaterais de linguagem
- Ausentes: apontadores, funções e procedimentos, campos, etc.

v1915

Semântica formal N, aula 2 – 14 / 30

Semântica operacional

15 / 30

Semântica operacional – Idéia

- A semântica operacional é uma definição formal da execução do IMP.
- A semântica abstrai aspectos da execução concreta, por exemplo endereços na memória.
- O significado de expressões depende dos valores das variáveis.

v1915

Semântica formal N, aula 2 – 16 / 30

IMP – categorias semânticas

Precisamos definir que os categorias sintáticas significam

- Um literal $n \in \text{Int}$ significa um número inteiro $\in \mathbb{Z}$
- $b \in \text{Bool}$ significa $\{\text{true}, \text{false}\}$
- Um local significa o valor neste local $\in \mathbb{Z}$
- Uma expressão booleana significa um valor booleana $\{\text{true}, \text{false}\}$
- Uma expressão aritmética significa uma número inteiro $\in \mathbb{Z}$
- Uma comando muda o estado (o valor das variáveis)
 - ◆ O estado atual é abstraído como uma função

$$\sigma \in \Sigma = [\text{Loc} \rightarrow \mathbb{Z}]$$

- ◆ Assim, o significado de um comando vai ser um novo estado $\in \Sigma$

v1915

Semântica formal N, aula 2 – 17 / 30

IMP – exemplo

Supondo um estado

$$\sigma = \{v \mapsto 5, w \mapsto 6\}$$

a expressão

$$2+v+w$$

significa intuitivamente 13. Escrevemos

$$2+v+w, \sigma \Downarrow 13$$

\Downarrow é uma relação de transição (entre $\text{AExp} \times \Sigma$ e \mathbb{Z})

Mas para chegar à essa conclusão precisamos regras! (Na verdade para os casos mais complicados)

v1915

Semântica formal N, aula 2 – 18 / 30

Regras

Escrevemos

$$\frac{P}{c}, \quad \frac{p_1 \dots p_n}{c} \quad \text{ou } P/c$$

para uma regra que permite a *conclusão* c com um conjunto de *premissas* $P = \{p_1, \dots, p_n\}$.
Se $P = \emptyset$ temos um *axioma* e escrevemos

$$\frac{}{c} \quad \text{ou só } c$$

Observa:

- A notação das regras foi inventada na lógica
- Aqui tratamos a sintaxe e permitimos também as meta-variáveis sintáticas

v1915

Semântica formal N, aula 2 – 19 / 30

Semântica das expressões aritméticas

$$\frac{}{\mathbf{n}, \sigma \Downarrow n} \text{ num}$$
$$\frac{}{\mathbf{l}, \sigma \Downarrow \sigma(l)} \text{ loc}$$
$$\frac{\mathbf{a}, \sigma \Downarrow n \quad \mathbf{a}', \sigma \Downarrow n'}{\mathbf{a+a'}, \sigma \Downarrow n''} \text{ sum, com } n'' = n + n'$$
$$\frac{\mathbf{a}, \sigma \Downarrow n \quad \mathbf{a}', \sigma \Downarrow n'}{\mathbf{a-a'}, \sigma \Downarrow n''} \text{ dif, com } n'' = n - n'$$

(A regra para \times é omitida. Exercício!)

Observe a diferença entre o símbolo $+$ e o operador $+$.

v1915

Semântica formal N, aula 2 – 20 / 30

Instâncias

Permitimos meta-variáveis nas regras. Se substituirmos as meta-variáveis, teremos *instâncias* das regras: Se σ tal que $\sigma(v) = 3$

$$\begin{array}{c}
 3, \sigma \Downarrow 3 \\
 v, \sigma \Downarrow 3 \\
 \hline
 5, \sigma \Downarrow 5 \quad v, \sigma \Downarrow 3 \quad \text{sum} \\
 \hline
 5+v, \sigma \Downarrow 8
 \end{array}$$

v1915

Semântica formal N, aula 2 – 21 / 30

Exemplo: expressões aritméticas

Com σ tal que $\sigma(i) = 3$ o que significa $(i+5) \times 2$?

$$\begin{array}{c}
 \frac{}{i, \sigma \Downarrow 3} \text{loc} \quad \frac{}{5, \sigma \Downarrow 5} \text{num} \\
 \hline
 i+5, \sigma \Downarrow 8 \quad \text{sum} \quad \frac{}{2, \sigma \Downarrow 2} \text{num} \\
 \hline
 (i+5) \times 2, \sigma \Downarrow 16 \quad \text{prod}
 \end{array}$$

v1915

Semântica formal N, aula 2 – 22 / 30

Equivalências

Escrevemos $a \equiv a'$ para expressões sintaticamente equivalentes.

$$5+3 \equiv 5+3$$

$$5+3 \not\equiv 3+5$$

Escrevemos $a \sim a'$ para expressões semânticamente equivalentes sse

$$\forall n \in \mathbb{Z}, \sigma \in \Sigma : a, \sigma \Downarrow n \leftrightarrow a', \sigma \Downarrow n.$$

$$5+3 \sim 5+3$$

$$5+3 \sim 3+5$$

v1915

Semântica formal N, aula 2 – 23 / 30

Semântica das expressões booleanas

$$\frac{}{\mathbf{t}, \sigma \Downarrow \mathbf{t}} \text{ bool}$$
$$\frac{\mathbf{a}, \sigma \Downarrow n \quad \mathbf{a}', \sigma \Downarrow n'}{\mathbf{a} < \mathbf{a}', \sigma \Downarrow \text{true}} \text{ le (se } n < n')$$
$$\frac{\mathbf{a}, \sigma \Downarrow n \quad \mathbf{a}', \sigma \Downarrow n'}{\mathbf{a} < \mathbf{a}', \sigma \Downarrow \text{false}} \text{ le (se } n \not< n')$$
$$\frac{\mathbf{b}, \sigma \Downarrow \text{true}}{\neg \mathbf{b}, \sigma \Downarrow \text{false}} \text{ not} \quad \frac{\mathbf{b}, \sigma \Downarrow \text{false}}{\neg \mathbf{b}, \sigma \Downarrow \text{true}} \text{ not}$$

(A regra para $=$ é omitida. Exercício!)

v1915

Semântica formal N, aula 2 – 24 / 30

Semântica das expressões booleanas...

$$\frac{\frac{\text{b}, \sigma \Downarrow t \quad \text{b}', \sigma \Downarrow t'}{\text{b} \wedge \text{b}', \sigma \Downarrow \text{true}} \wedge (\text{se } t = \text{true e } t' = \text{true})}{\frac{\text{b}, \sigma \Downarrow t \quad \text{b}', \sigma \Downarrow t'}{\text{b} \wedge \text{b}', \sigma \Downarrow \text{false}} \wedge (\text{se } t = \text{false ou } t' = \text{false})}$$

(A regra para \vee é omitida. Exercício!)

v1915

Semântica formal N, aula 2 – 25 / 30

Exemplo: expressões booleanas

Com σ tal que $\sigma(i) = 3$ o que significa $\neg(i < 0) \wedge (i < 4)$?

$$\frac{\frac{\frac{\frac{\text{loc } i, \sigma \Downarrow 3}{i < 0, \sigma \Downarrow \text{false}}{\neg(i < 0), \sigma \Downarrow \text{true}} \text{ not} \quad \frac{\frac{\frac{\text{loc } i, \sigma \Downarrow 3 \quad \text{num } 4, \sigma \Downarrow 4}{i < 4, \sigma \Downarrow \text{true}} \text{ le}}{\neg(i < 0) \wedge (i < 4), \sigma \Downarrow \text{true}} \wedge$$

v1915

Semântica formal N, aula 2 – 26 / 30

Semântica dos comandos...

$$\begin{array}{c}
 \frac{}{\text{skip}, \sigma \Downarrow \sigma} \text{ skip} \\
 \frac{}{\text{a}, \sigma \Downarrow n} \text{ assign} \\
 \frac{}{\text{x} := \text{a}, \sigma \Downarrow \sigma[x \mapsto n]} \text{ assign} \\
 \frac{\text{c}, \sigma \Downarrow \sigma' \quad \text{c}', \sigma' \Downarrow \sigma''}{\text{c}; \text{c}', \sigma \Downarrow \sigma''} \text{ seq} \\
 \frac{\text{b}, \sigma \Downarrow \text{true} \quad \text{c}, \sigma \Downarrow \sigma'}{\text{if b then c else c}', \sigma \Downarrow \sigma'} \text{ if} \\
 \frac{\text{b}, \sigma \Downarrow \text{false} \quad \text{c}', \sigma \Downarrow \sigma'}{\text{if b then c else c}', \sigma \Downarrow \sigma'} \text{ if}
 \end{array}$$

v1915

Semântica formal N, aula 2 – 27 / 30

Semântica dos comandos...

$$\begin{array}{c}
 \frac{\text{b}, \sigma \Downarrow \text{false}}{\text{while b do c}, \sigma \Downarrow \sigma} \text{ while} \\
 \frac{\text{b}, \sigma \Downarrow \text{true} \quad \text{c}; \text{while b do c}, \sigma \Downarrow \sigma'}{\text{while b do c}, \sigma \Downarrow \sigma'} \text{ while}
 \end{array}$$

v1915

Semântica formal N, aula 2 – 28 / 30

Exemplo: Comandos

Com σ tal que $\sigma(i) = 3$ o que significa $i := i+1; \text{skip}; i := i*2$?

$$\frac{\frac{\frac{i+1, \sigma \Downarrow 4}{\text{assign}}}{i := i+1, \sigma \Downarrow \sigma[i \mapsto 4]} \quad \frac{\text{skip}, \sigma[i \mapsto 4] \Downarrow \sigma[i \mapsto 4]}{\text{seq}} \quad \frac{i*2, \sigma[i \mapsto 4] \Downarrow 8}{\text{assign}}}{i := i+1; \text{skip}, \sigma \Downarrow \sigma[i \mapsto 4]} \quad \frac{i := i*2, \sigma[i \mapsto 4] \Downarrow \sigma[i \mapsto 8]}{\text{seq}}}{i := i+1; \text{skip}; i := i*2, \sigma \Downarrow \sigma[i \mapsto 8]} \text{seq}$$

v1915

Semântica formal N, aula 2 – 29 / 30

Discussão e resumo

- As regras estão parecido com as regras lógicas da dedução natural
- A regras seguem o conceito do valor de um fragmento: dado os valores de outras fragmentos, o valor de fragmento em consideração pode ser concluído em um passo
- Por isso, esse semântica se chama *semântica operacional natural* ou *semântica de passos grandes* (inglês: big-step semantics)

O que temos agora?

- Temos uma especificação formal, não ambigua para IMP.
- A regras permitem provar características de IMP (vamos ver!)
- A regras permitem uma implementação fácil de um interpretador.

v1915

Semântica formal N, aula 2 – 30 / 30