Fundamentos de Algoritmos - Aula 1

CIC/UFRGS

2006/1

Números e Aritmética

- espécies mais populares
 - inteiros positivos
 - inteiros negativos
 - frações, ou racionais
 - reais

5 -5 2/3 17/3 #i1.4142135623731

(representação inexata de real)

- programador pode fazer
 - somas: $(-5\ 5)\ +$
 - subtrações: $(- 5\ 5)$
 - multiplicações: $(*\ 3\ 4)$
 - divisões: $(/\ 8\ 12)$
- toda expressão é parentizada e o operador vem no início seguido dos números separados por espaço

Avaliação é tal como nós fazemos

$$\begin{aligned} & (* (+ 2 2) (/ (* (+ 3 5) (/ 30 10)) 2)) \\ = & (* 4 (/ (* 8 3) 2)) \\ = & (* 4 (/ 24 2)) \\ = & (* 4 12) \\ = & 48 \end{aligned}$$

- por causa da forma, nunca há dúvidas sobre ordem da avaliação
- compare com $3 + 4 * 5$
- com tempo e prática aprendemos que a multiplicação é feita primeiro e depois a soma

Outras operações:

- `(sqrt A)` computa $\sqrt{A}$
- `(expt A B)` computa A^B
- `(remainder A B)` computa o resto da divisão inteira A/B
- `(log A)` computa logaritmo natural de A
- `(sin A)` computa o seno de A (em radianos)

- Scheme computa com inteiros e racionais EXATOS, se usarmos operações primitivas que produzem resultados exatos
- assim o resultado de $(/ 44 14)$ é exibido como $22/7$
- Scheme (e outras linguagens) produzem valores INEXATOS quando lidam com números reais
- raiz quadrada de 2 não é racional e sim um real

A notação `i` avisa o programador que o resultado é uma aproximação do valor verdadeiro

```
(- #i1.0 #i0.9)
= #i0.0999999999999999999998
```

mas

```
(- #i1000.0 #i999.9)
= #i0.100000000000000002274
```

mesmo sabendo, da matemática, que ambas diferenças deveriam ser 0, 1

Na disciplina: números exatos

Variáveis e Programas

- em matemática aprendemos a formular dependências entre quantidades usando VARIÁVEIS que representa uma quantidade desconhecida
- exemplo: um disco de raio r tem área aproximada de $3,14.r^2$
- para um disco de raio 5 fazemos

$$3,14.5^2 = 3,14.25 = 78,5$$

- uma expressão com variáveis é uma regra que decreve como computar um número quando temos valores para as variáveis
- um programa é uma regra que diz como produzir um dado a partir de outro dado
- deve haver uma maneira de nomear regras
- regra para computar área de um disco

```
(define (area-of-disk r)  
  (* 3.14 (* r r)))
```

- depois de definida, uma regra pode ser usada
- fornecendo um valor para cada variável que segue o nome da regra
- `(area-of-disk 5)`
- dizemos que **APLICAMOS** `area-of-disk` para 5.

```
(area-of-disk 5)
= (* 3.14 (* 5 5))
= (* 3.14 25)
= 78.5
```

- muitos programas consomem mais do que uma entrada
- programa para computar a área de um anel, ou seja de de um disco com um buraco no meio
- área do anel é a área do disco externo menos a área do disco interno
- há portanto dois valores desconhecidos: o raio do disco interno e o raio do disco externo

```
(define (area-of-ring outer inner)
  (- (area-of-disk outer)
     (area-of-disk inner)))
```

- `area-of-ring` é um programa
- que aceita duas entradas, chamadas `outer` e `inner`
- resultado será a diferença entre *(area-of-disk outer)* e *(area-of-disk inner)*
- usamos operações básicas de Scheme e também programas definidos

Para usar *area-of-ring* fornecemos duas entradas

```
(area-of-ring 5 3)
```

Essa expressão é a avaliada da seguinte forma

```
(area-of-ring 5 3)
```

```
= (- (area-of-disk 5)  
      (area-of-disk 3))
```

```
= (- (* 3.14 (* 5 5))  
      (* 3.14 (* 3 3)))
```

= . . .

Enunciado de Problemas

- raramente o problema é representar expressões aritméticas em programas
- descrições de problemas em geral são textos informais com ambigüidades, detalhes sem importância
- primeira tarefa: obter a informação relevante e só então formular a expressão apropriada

Eis um exemplo

A empresa XYX & Co. paga aos seus empregados \$12 por hora. Um empregado típico trabalha de 20 a 65 horas por semana. Desenvolva um programa que alcule o salário de um empregado a partir do seu número de horas trabalhadas

Se um empregado trabalha h horas, o seu salário será $12 \cdot h$. O programa será

```
(define (wage h)
  (* 12 h))
```

Erros de Sintaxe

- expressões são ou ATÔMICAS (numeros e variáveis)
- ou são expressões COMPOSTAS (que começam com “(”, seguido de operação, mais expressões e terminada por “)”)
- definições tem a seguinte forma:

```
(define (f x ... y)
  an-expression)
```

As seguintes definições possuem erros de sintaxe:

```
(define (P x)
  (+ (x) 10))
```

```
(define (Q x)
  x 10)
```

Erros de Execução

- nem toda expressão legal possui um resultado
- um exemplo óbvio é `(/ 1 0)` .
- da mesma forma, se definimos

```
(define (f n)
  (+ (/ n 3) 2))
```

DrScheme não pode avaliar `(f 5 8)`

- RUN-TIME ERROR

Erros Lógicos

- programador pode também cometer ERROS LÓGICOS
- erros lógicos não são detectados por DrScheme e nenhuma mensagem é produzida
- um programador pode evitar tais erros somente projetando seus programas com cuidado e com método
- exemplo de programa com erro lógico

```
(define (wage h)
  (+ 12 h))
```

Compondo Funções

- em geral programa consiste de várias definições
- programa para **cálculo da área de anel** possui 2 definições: *area-of-ring* e *area-of-disk*
- *area-of-ring* é a FUNÇÃO PRINCIPAL,
- *area-of-disk* é FUNÇÃO AUXILIAR

- Eis duas versões para o cálculo da área de um anel.
- Qual das duas é mais legível?

```
(define (area-of-ring outer inner)
  (- (area-of-disk outer)
     (area-of-disk inner)))
```

e

```
(define (area-of-ring outer inner)
  (- (* 3.14 (* outer outer))
     (* 3.14 (* inner inner))))
```

- para programas pequenos a diferença entre os dois estilos é pequena
- para programas maiores o uso de funções auxiliares é uma necessidade
- mas mesmo para programas menores devemos considerar dividir o problema com o uso de auxiliares

Exercícios

1. Leia os capítulos 1, 2 e 3 do livro www.htdp.org
2. Instale o DrScheme em seu computador e/ou use o DrScheme já instalado nos laboratórios
3. Experimente com o DrScheme
 - Experimente com algumas funções pré-definidas
 - Use a avaliação passo a passo (*stepper*)
 - Force erros de sintaxe e de execução e observe as mensagens de erro produzidas pelo DrScheme