

Interpretação e Compilação de Linguagens de Programação

Exame de Época Normal 2010-2011

4 de Janeiro de 2011

Notas: O exame pode ser realizado com consulta de materiais em suporte papel trazidos pelo aluno. O exame tem a duração de 3h.

1. Considere a linguagem *OnO* cuja sintaxe concreta é definida pela seguinte gramática:

$$\begin{array}{l} E ::= [Id_1 = E_1, \dots, Id_n = E_n, l_1(Id) = E'_1, \dots, l_m(Id) = E'_m] \\ \quad | \quad Num \quad | \quad E + E \quad | \quad Id \\ \quad | \quad E.l(E) \quad | \quad Id := E \quad | \quad self \\ \quad | \quad decl \quad Id = E \quad in \quad E \end{array}$$

As construções básicas da linguagem *OnO* são os números inteiros (*Num*) e as operações habituais sobre os números inteiros (aqui simbolizadas pela operação de adição), os identificadores (*Id*), e a declaração de objectos $[Id_1 = E_1, \dots, Id_n = E_n, l_1(x) = E'_1, \dots, l_m(x) = E'_m]$. Os objectos são definidos por um conjunto de variáveis de instância, com os identificadores Id_1, \dots, Id_n , inicializados com os valores denotados pelas expressões E_1, \dots, E_n , e por um conjunto de métodos com nomes l_1, \dots, l_m , um parâmetro e as expressões E'_1, \dots, E'_m . Para além disso a linguagem tem as expressões de chamada de um método $E.l(E)$, a afectação de uma variável de instância, a declaração de um identificador (*decl*), e um identificador especial *self* que denota o próprio objecto. A linguagem *OnO* tem passagem de parâmetros por valor (*call-by-value*) e resolução estática de nomes.

Considere o seguinte exemplo da linguagem *OnO*.

```
decl cell = [new(x) = [a = x, set(x) = a := x, get(x) = a]] in
decl point = [x = cell.new(0), y = cell.new(0),
              moveX(d) = x.set(x.get(0) + d),
              moveY(d) = y.set(y.get(0) + d)] in
point.moveX(10)
```

- (a) [1 valor] **Descreva** a sintaxe abstracta da linguagem *OnO* usando um conjunto de classes Java.
Nota: Apresente apenas a assinatura de um constructor de cada classe com o seu nome e um parâmetro para cada uma das suas variáveis de instância.
- (b) [1 valor] **Represente** os valores da linguagem *OnO* numa hierarquia de classes Java.
- (c) [4 valores] **Defina** a semântica para as construções da linguagem *OnO* através da implementação do método **evaluate** de cada uma das classes correspondentes.
Explique sucintamente quais os parâmetros e quais os possíveis valores de retorno do método.
- (d) [1 valor] **Enumere** quais os erros de execução que podem ocorrer num programa da linguagem *OnO*.

2. Considere o seguinte exemplo na linguagem **Core** definida nas aulas com resolução estática de nomes:

```
decl x = 1 in
decl f = fun y => fun z => x + y + z end end end in
decl g = fun x => f(x + 1) in
(g(3))(4)
```

- (a) [2 valores] De acordo com o estudado nas aulas da disciplina, **indique qual o ambiente de compilação** da expressão $x + y + z$, e quais **os endereços** dos identificadores livres na expressão.
- (b) [1 valor] **Indique** qual seria o resultado do programa no caso de ser interpretado com resolução dinâmica de nomes.
- (c) [1 valor] **Justifique** se é possível definir um procedimento de compilação para a linguagem **Core** se esta for definida com resolução dinâmica de nomes.
3. A utilização de uma pilha de execução como modelo de base para a execução impõe, em geral, restrições à linguagem de programação utilizada.
- (a) [1 valor] **Apresente e justifique através de exemplos de linguagens existentes** algumas dessas restrições.
- (b) [1 valor] **Justifique** se é possível definir um procedimento de compilação da linguagem *OnO* para CIL que utilize apenas o suporte nativo para objectos, e funções de objectos?
4. Considere a linguagem *ToNo*, uma extensão tipificada da linguagem *OnO* com anotações de tipos na declaração dos métodos de objectos e com um tipo especial *Self* que denota o tipo do próprio objecto.

$$E ::= [Id_1 = E_1, \dots, l_1(x:T_1):T'_1=E'_1, \dots, l_m(x:T_m):T'_m=E'_m] \dots$$

- (a) [1 valor] **Indique** quais os tipos necessários para tipificar a linguagem *ToNo*.
- (b) [3 valores] **Defina** o sistema de tipos da linguagem *ToNo* através de um conjunto de regras de tipos.
- (c) [1 valor] **Apresente** a versão tipificada do programa *OnO* da pergunta 1.
- (d) [2 valores] Dadas as declarações para os identificadores *point* e *cell* do programa exemplo da pergunta 1. **Apresente** a tipificação da expressão *point.moveX(cell.new(0).get(0))*, através de uma derivação de tipos.