

Subprogramas

Gustavo Scaloni Vendramini
Guilherme José Henrique
Sean Carlisto de Alvarenga
Vinícius Fernandes de Jesus

22 de setembro de 2013

<i>SUMÁRIO</i>	2
----------------	---

Sumário

1 Subprogramas Como Parâmetro	3
--------------------------------------	----------

1 Subprogramas Como Parâmetro

Em muitas ocasiões temos a necessidade de passar um subprograma através de um parâmetro. A ideia é interessante e simples, mas gera duas complicações em termos de implementação.

Primeiro, temos a complicação que consiste na maneira de realizar a checagem de tipo (*type checking*, `??`) do subprograma passado por parâmetro. Em C e C++, onde a passagem de subprogramas é feita através de ponteiro para função, essa checagem é feita pelo tipo do ponteiro.

A outra complicação ocorre em linguagens de programação que permitem subprogramas aninhados. O problema refere-se a qual ambiente de referência o subprograma passado por parâmetro terá. Nessa situação, há três tipos possíveis:

Shallow Binding: O ambiente é o local onde o subprograma é chamado.

Deep Binding: O ambiente refere-se onde o subprograma foi definido.

Ad Hoc Binding: O ambiente condiz com o local que o subprograma foi passado por parâmetro.

Como exemplo, considere a listagem 1, cuja syntax é de JavaScript. O subprograma `sub2()` apenas imprime o valor da variável `x`, porém, seu valor depende do ambiente de referência utilizado.

```
1 function sub1() {  
2   var x;  
3   function sub2() {  
4     alert(x);  
5   };  
6   function sub3() {  
7     var x;  
8     x = 3;  
9     sub4(sub2);  
10  };  
11  function sub4(subx) {  
12    var x;  
13    x = 4;  
14    subx();  
15  };  
16  x = 1;  
17  sub3();  
18 };
```

Listing 1: Código retirado de [1]

Caso a listagem em questão utilize o ambiente de referência *Shallow Binding*, o valor impresso seria 4. Caso o ambiente *Deep Binding* fosse escolhido, o valor impresso seria 1. Já para *Ad Hoc Binding*, o valor seria 3.

Referências

- [1] Robert W. Sebesta. *Concepts of Programming Languages (10th Edition)*. Addison-Wesley, 2012.