

Lista II - Linguagens de Programação

José Douglas Gondim Soares, 485347

1- Se a variável no local de armazenamento referenciada por "P" for "p", então "p" é uma variável ligada ao valor do procedimento.

2- Sem este passo, o comportamento de proc MulByN mudaria após cada chamada da função. Este passo garante que a execução seja determinada apenas pelos seus argumentos.

• Exemplo em que N não existe no ambiente:

declare A, B

A = 10

local MulByN in

local N in

N = 3

proc {MulByN x ?y}

y = x * N

end

end

{MulByN A B}

% N não existe

end

• Um exemplo em que N existe no ambiente no momento da chamada, mas está ligado a um valor diferente de 3

```
declare A B
```

```
A = 10
```

```
local MulByN in
```

```
  local N in
```

```
    N = 3
```

```
    Proc {MulByN x ?y}
```

```
      Y = x * N
```

```
    end
```

```
  end
```

```
local N in
```

```
  N = 10
```

```
  {MulByN A B}
```

```
  % N = 10
```

```
end
```

```
end
```

(13-) declare x y z w | Para resolver as dúvidas ao lado, basta
x = [a z] | imaginar que estas são apenas instruções
y = [w b] | de como essas variáveis devem ser
x = y | ligados. Dessa forma, qualquer ordem que
| eu declarar essas relações resultará no
| mesmo estado final

4- a)

```
case Y  
of false then  
  <stm> 2
```

```
[] else  
  <stm> 1
```

```
end
```

b)

```
if {Label <y>} == <label> then
```

```
  if {Arity <y>} == [<feature> 1 ... <feature> n] then
```

```
    local <y>1 = <y>. <feature> 1 ... <y> n = <y>. <feature> n in  
    <stm> 1
```

```
  end
```

```
  else <stm> 2
```

```
end
```

```
else <stm>
```

```
end
```

5- {Test [b c a]} % imprime 4

{Test f(b(3))} % imprime 5

{Test f(a)} % imprime 2

{Test f(a(3))} % imprime 5

{Test f(d)} % imprime 5

• {Test [a,b,c]} % imprime 4

{Test [c a b]} % imprime 4

{test a1a} % imprime 1

{Test "1"(a b c)} % imprime 6

Obs: basta olhar a estrutura de cada entrada e ver em qual dos casos ela é melhor representada. Se nenhuma é representativa, imprime o else (caso b).

7- No caso $\{ \text{Max} \ 3 \ 4 \}$, o value assumirá valor 3 e o X, 4. Como $4 > 3 == \text{true}$, será retornado o valor de X que é 4.

No caso $\{ \text{Max} \ 5 \ 4 \}$, $\text{Value} = 5$, $X = 4$. Logo $5 > 4 == \text{true}$ e S é retornado.

portanto, será impresso $[4 \ 5]$.

8- a) Elas dão exatamente o mesmo resultado. Quando BP1 é falsa, BP2 nunca é chamada.

b)

```
fun{OnElse BP1 BP2}
  if {BP1 == true} then true
  else {BP2}
end
```

end

| Se BP1 for verdadeira, a função já retorna verdadeira, caso contrário, ela retorna o valor da BP2, que pode ser verdadeira ou falsa.

11- podemos observar que cada função só é chamada ao final da outra, o que leva a um código sequencial onde eu não me preocupo com o estado anterior. Logo essas funções executam com um tamanho de pilha constante, uma vez que uma não espera o resultado da outra para retornar.