

Expressão **case**

Expressão **case** é uma forma de expressão que realiza **casamento de padrão** entre o valor de uma **expressão** e um ou mais **padrões**.

Uma expressão case é da forma:

```
case exp of  
  padrao1 ➔ res1  
  ...  
  padraon ➔ resn
```

onde

- *exp*, *res*₁, ..., *res*_{*n*} são **expressões**
- *padrao*₁, ..., *padrao*_{*n*} são **padrões**
- *exp*, *padrao*₁, ..., *padrao*_{*n*} devem ser do **mesmo tipo**
- *res*₁, ..., *res*_{*n*} devem ser do **mesmo tipo**, que determina o **tipo** da expressão **case**
- os padrões devem estar alinhados na mesma coluna (regra de *layout*)

Avaliação de expressões case

É feito o **casamento de padrão** do valor de *exp* com os padrões, na seqüência em que foram escritos, até que se obtenha sucesso ou se esgotem os padrões

O **primeiro** padrão cujo casamento suceder é escolhido

O **resultado** final da expressão case é dado pela expressão associada ao padrão escolhido

O resultado é avaliado em um **ambiente estendido** com as associações de variáveis resultantes do casamento de padrão

Se a expressão **não casar** com nenhum padrão, a avaliação da expressão case resulta em um **erro**

Exemplos de expressões case

A expressão

```
case 3-2+1 of  
0  => "zero"  
1  => "um"  
2  => "dois"  
3  => "tres"
```

resulta em "dois", pois o valor da expressão $3-2+1$ é 2, que casa com o terceiro padrão 2, selecionando "dois" como resultado.

Exemplos de expressões case (cont.)

A expressão

```
case 23>10 of
  True  -> "beleza!"
  False -> "oops!"
```

resulta em "beleza!", pois o valor da expressão $23 > 10$ é **True**, que casa com o primeiro padrão **True**, seleccionando "beleza!" como resultado.

Exemplos de expressões case (cont.)

A expressão

```
case toUpper (head "masculino") of  
  'F' -> 10.2  
  'M' -> 20.0
```

resulta em 20.0, pois o valor da expressão `toUpper(head "masculino")` é `'M'`, que casa com o segundo padrão `'M'`, selecionando 20.0 como resultado.

Exemplos de expressões case (cont.)

A expressão

```
case head "masculino" of  
  'F' -> 10.2  
  'M' -> 20.0
```

resulta em um erro em tempo de execução, pois o valor da expressão `head "masculino"` não casa com nenhum dos padrões.

Exemplos de expressões `case` (cont.)

A expressão

```
case toupper (head "masculino") of  
  'F' -> "mulher"  
  'M' -> 20.0
```

está incorreta, pois os resultados `"mulher"` e `20.0` não são do mesmo tipo.

Exemplos de expressões `case` (cont.)

A expressão

```
case head "Masculino"=='F' of
  True  -> "mulher"
  1     -> "homem"
```

está incorreta, pois os padrões **True** e **1** não são do mesmo tipo.

Exemplos de expressões case (cont.)

A expressão

```
case head "Masculino" of  
  True  -> "mulher"  
  False -> "homem"
```

está incorreta, pois a expressão `head "Masculino"` e os padrões `True` e `False` não são do mesmo tipo.

Exemplos de expressões case (cont.)

A expressão

```
case toUpper (head "masculino") of
  'F' → 10.0
  'M' → 20.0
```

está incorreta, uma vez que não segue a regra de *layout* (os padrões não estão na mesma coluna).

Exemplos de expressões case (cont.)

A expressão

```
case 3-2+1 of
  x -> 11*x
```

resulta em 22, pois o valor da expressão $3-2+1$ é 2, que casa com o primeiro padrão x , associando a variável x com o valor 2, e selecionando $11*x$ como resultado

Definições locais com **where**

A palavra reservada **where** é utilizada para introduzir definições locais cujo contexto (ou **escopo**) é a expressão no lado direito e as guardas (quando houver) de uma equação.

Exemplo:

```
f(x,y) = (a+1)**(a+2) where a = (x+y)/2
```

```
f(2,3) ~ 15.75
```

```
f(5,1) ~ 20.0
```

Definições locais com **where**

(cont.)

Quando há duas ou mais definições locais, elas podem ser escritas em diferentes estilos. **Exemplos:**

```
f(x,y)  = (a+1)*(b+2)
where { a = (x+y)/2; b = (x+y)/3 }
```

```
f(x,y)  = (a+1)*(b+2)
where a = (x+y)/2; b = (x+y)/3
```

```
f(x,y)  = (a+1)*(b+2)
where
    a = (x+y)/2
    b = (x+y)/3
```

Neste último exemplo foi usada a regra de *layout*, que dispensa os símbolos `;`, `{` e `}` mas exige que cada definição local esteja alinhada em uma mesma coluna.

Definições locais com **where**

(cont.)

Exemplo:

```
square x = x*x

g x y | x <= 10 = x+a
      | x > 10  = x-a

where
  a = square (y+1)
```

O escopo de **a** inclui os dois possíveis resultados, determinados pelas guardas.

Definições locais com **where**

(cont.)

As definições locais podem ser de funções e de variáveis, fazendo uso de padrões.

Exemplo:

```
h y = 3 + f y + f a + f b
  where
    c = 10
    (a,b) = (3*c, f 2)
    f x = x+7*c
```

```
h 5 ~ 320
```

O escopo de **a** inclui os dois possíveis resultados, determinados pelas guardas.

Definições locais com **where**

(cont.)

Definições locais com **where** não são compartilhadas entre corpos de funções de diferentes padrões nas equações.

Exemplo:

```
audacao :: String -> String
audacao "Joana" = saudacaoLegal ++ "Joana!"
audacao "Fernando" = saudacaoLegal ++ "Fernando!"
audacao nome = saudacaoInfeliz ++ " " ++ nome

where
    saudacaoLegal = "Ola! Que bom encontrar voce,"
    saudacaoInfeliz = "Oh! Pfft. E voce,"
```

Esta definição de função está incorreta. Para corrigi-la, transforme as definições locais de `saudacaoLegal` e `saudacaoInfeliz` em definições globais.