Decisões binárias e múltiplas

Programação (PG)

Centro de Cálculo
Instituto Superior de Engenharia de Lisboa

Pedro Alexandre Pereira (palex@cc.isel.ipl.pt)

Instrução / Expressão

Instrução

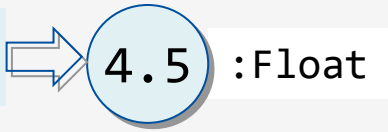
- É executada
- Realiza uma ação

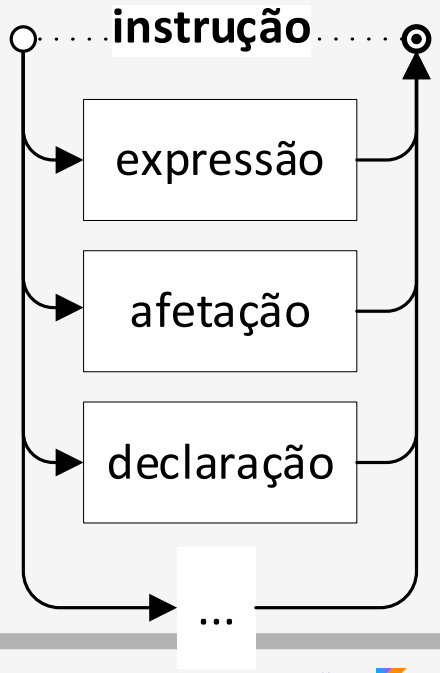
```
println("Hello")
```

Expressão

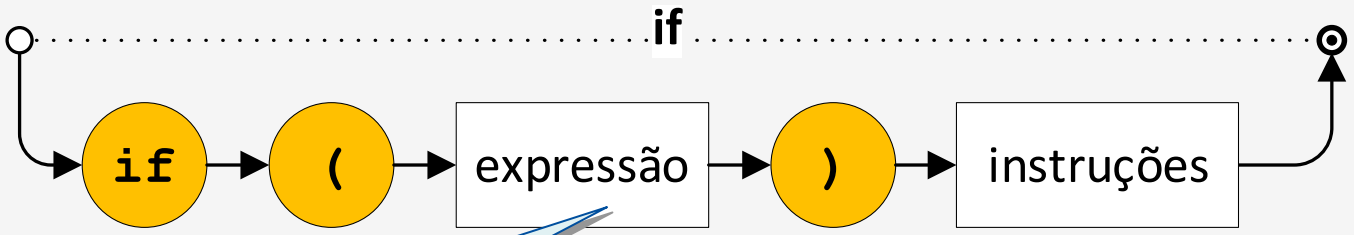
- É avaliada
- Produz um valor
  - De um tipo determinado

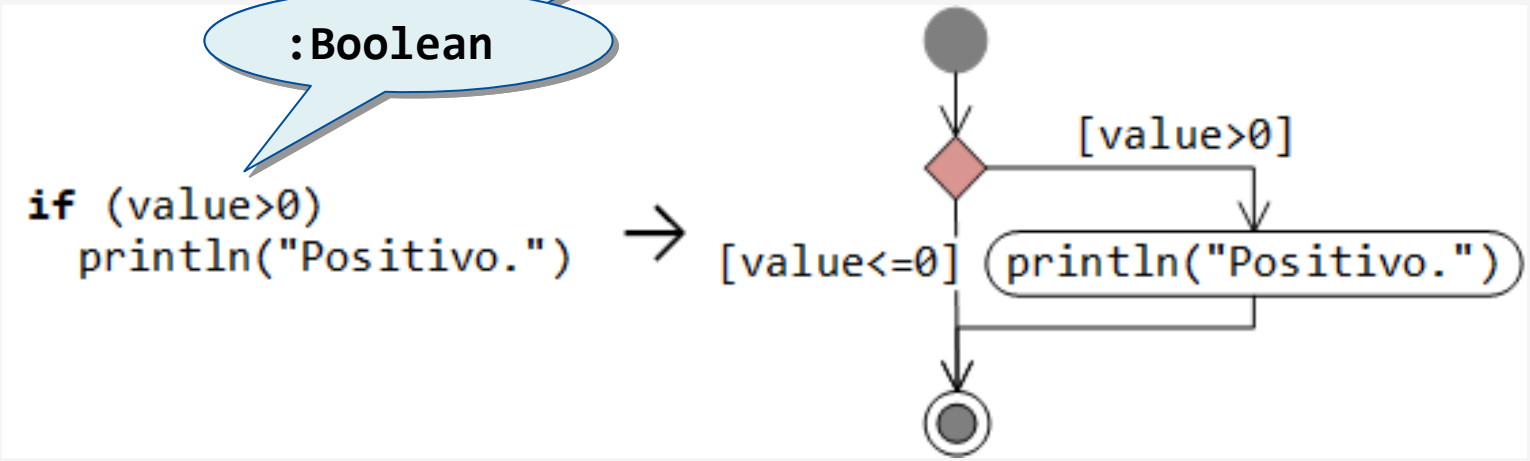
```
45f / 10
```

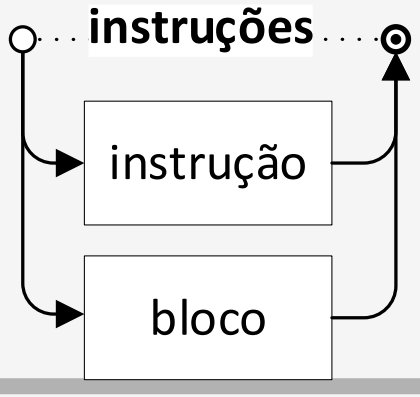


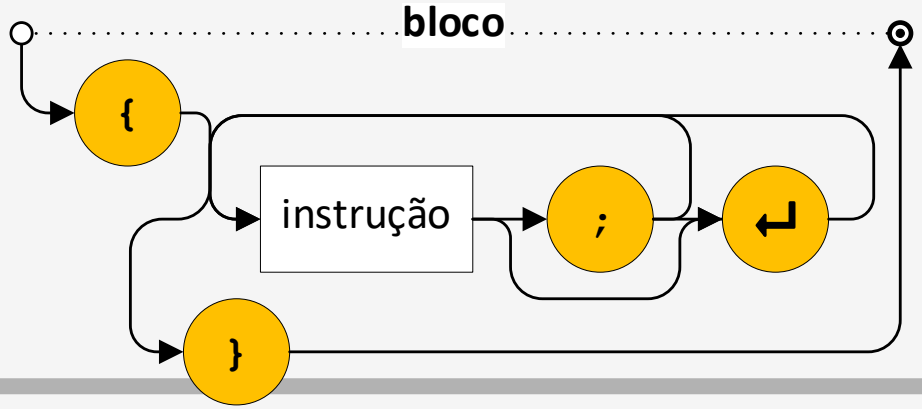


Instrução `if` (versão mais simples)

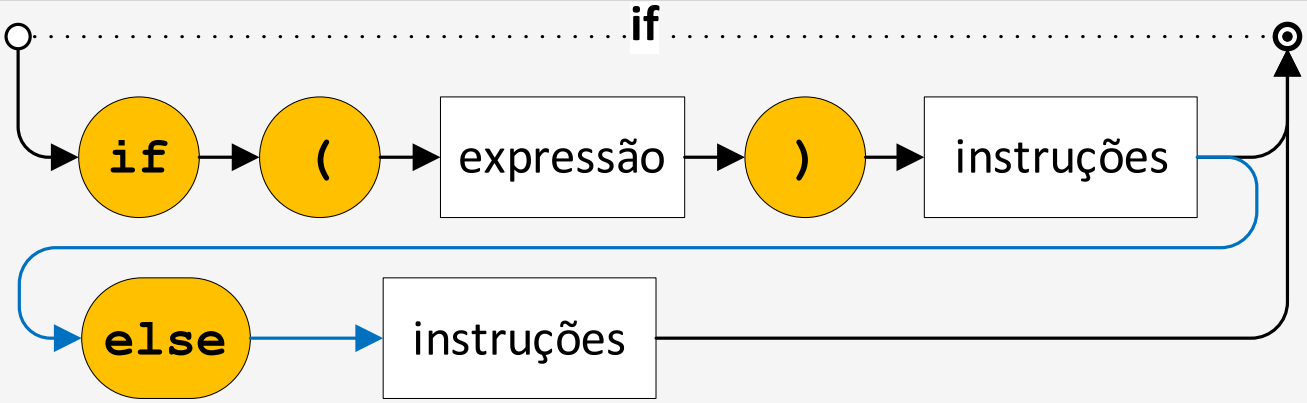




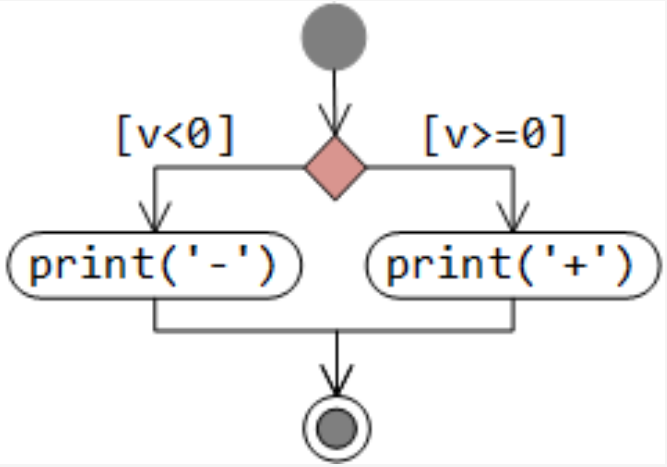




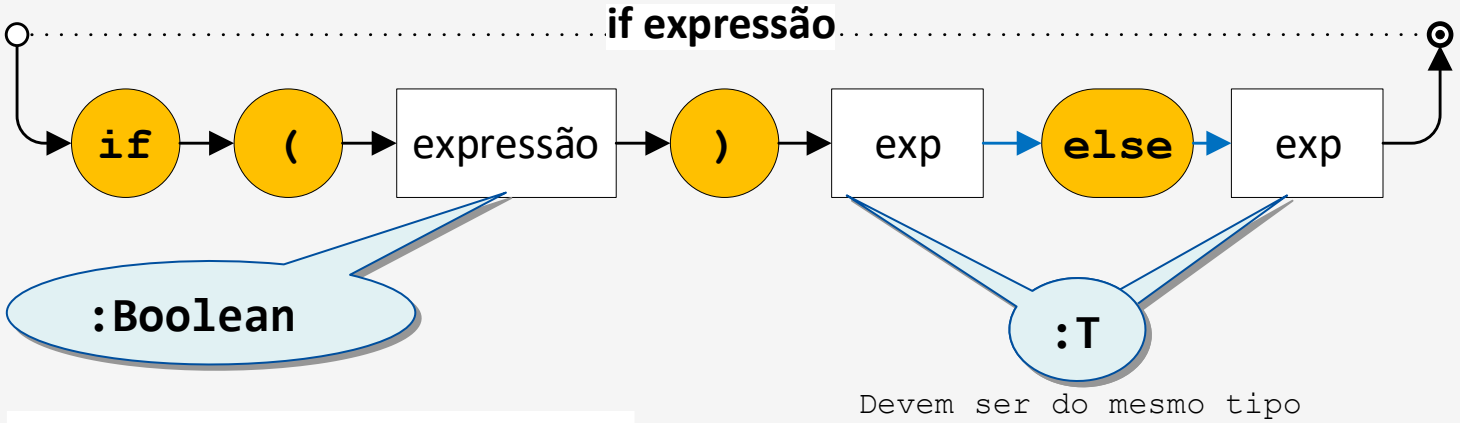
Instrução `if` (com `else`)




```
if (v>=0)
   print('+')
else
   print('-')
```

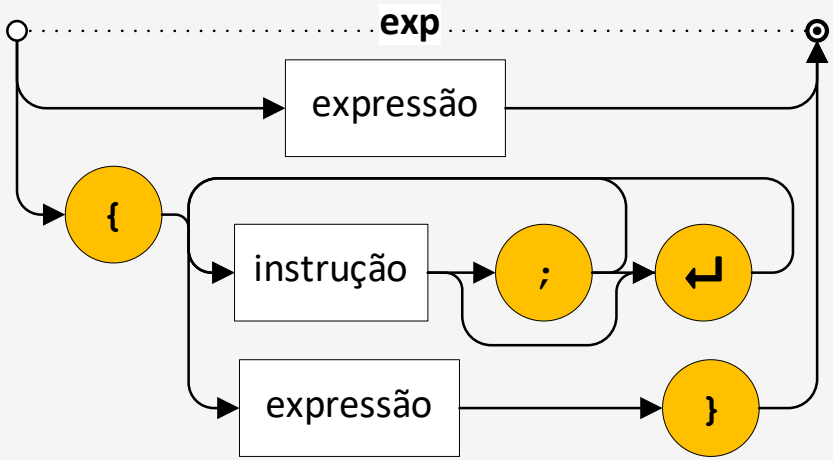


Expressão `if`

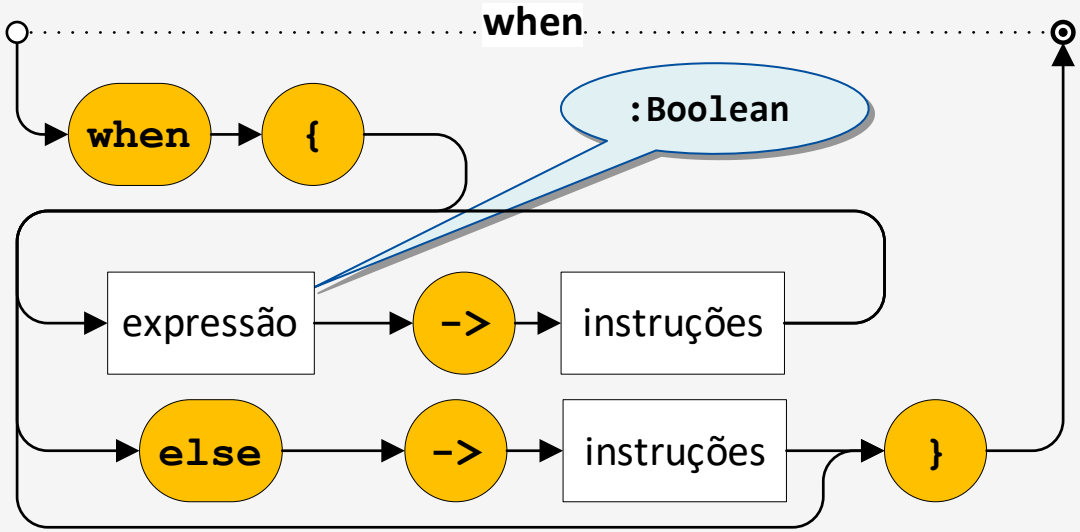



```
println( if ( a > b ) a else b )
```

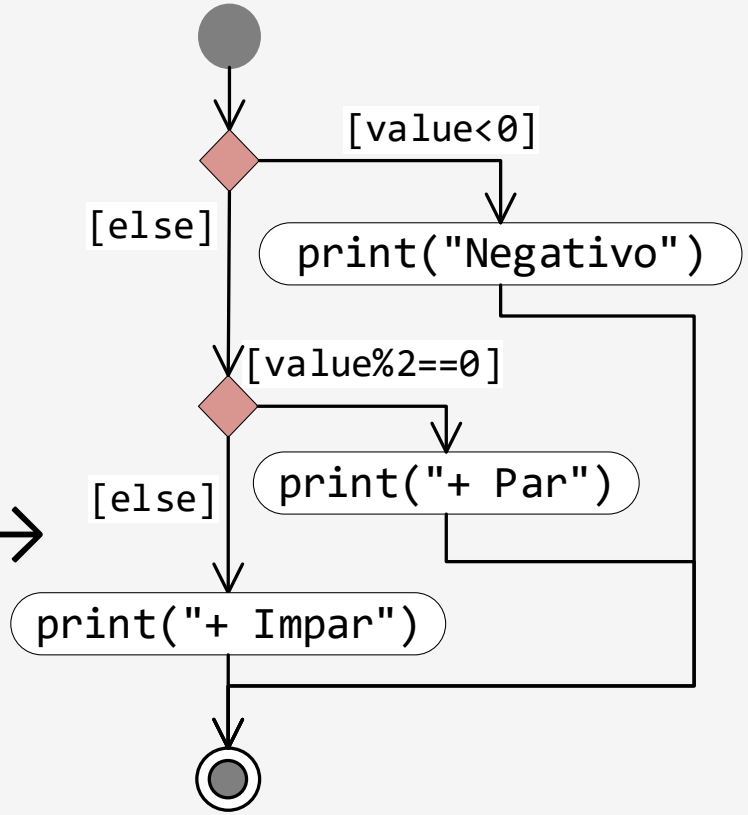
```
val factor = if ( a > 0 ) {
    print("positivo")
    +1
} else {
    print("negativo")
    -1
}
```



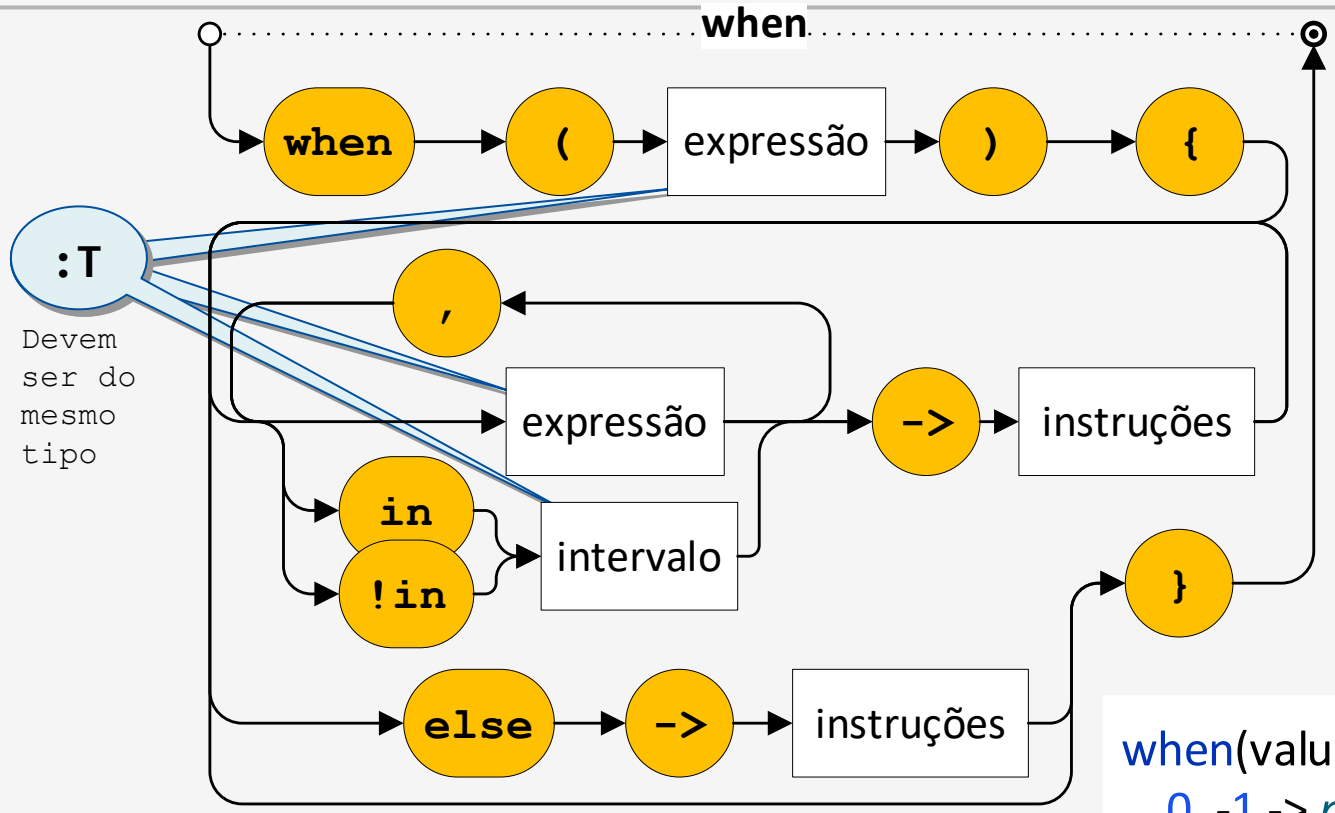
Instrução **when** (versão mais simples)




```
when {
   value < 0   -> print("Negativo")
   value % 2 == 0 -> print("+ Par")
   else        -> print("+ Impar")
}
```

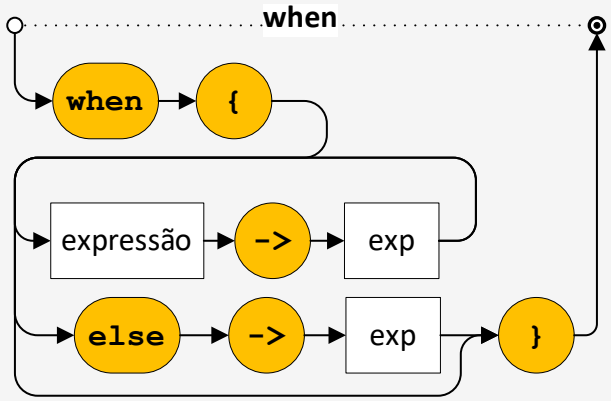


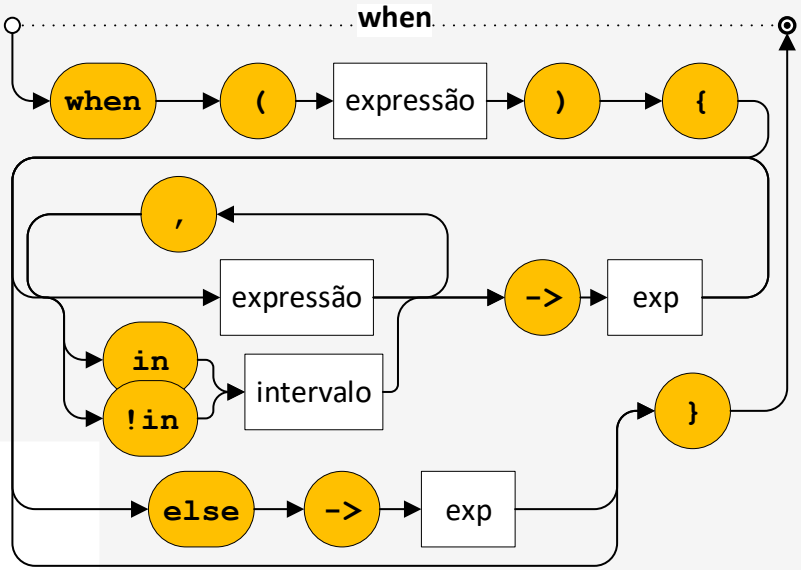
Instrução `when` com expressão a considerar




```
when(value) {
    0, -1 -> println("A")
    in 1..10 -> println("B")
    !in 15..20 -> println("C")
    else -> println("Outros")
}
```

Expressão **when**






```
val daysOfMonth = when(month) {
   2 -> {
      print("Ano? ")
      val y = readInt()
      if (y%4==0 && y%100!=0 || y%400==0) 29 else 28
   }
   4, 6, 9, 11 -> 30
   else -> 31
}
```

```
println(
   when(value) {
         0,2,4,6,8 -> 2
         else -> 1
   }
)
```

- Os ramos têm que abranger todos os possíveis valores
  - Na maioria dos casos obriga a ter o ramo else