# Report #8 - Dataflow Coverage

## Introdução

Dataflow testing é uma estratégia de software que é utilizada de modo a manter o foco nas variáveis utilizadas pelo programa e os seus valores. Utilizando fluxogramas para descrever os caminhos possíveis por parte do sistema, permitindo assim testar os caminhos possíveis e suas ramificações, que são analisadas utilizando um variado conjunto de tabelas:

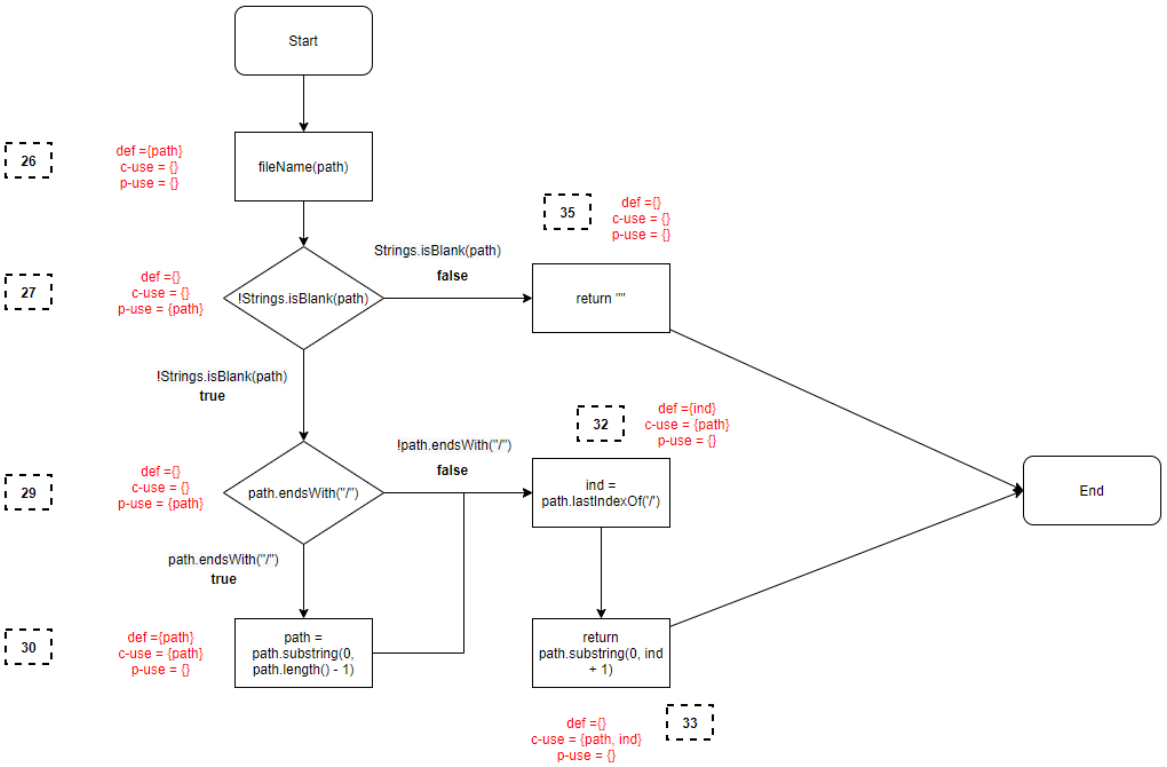
- **p-use:**
    - predicate use, servindo como um valor utilizado num predicado booleano condicional.
- **c-use:**
    - computation use, servindo como um local onde o valor de uma variável é utilizado fora do contexto booleano, como atribuições e cálculos aritméticos.
- **def-use:**
    - def-use é correspondente a identificação de pares (d,u) em que estes representam nós/arestas onde uma variável é utilizada;
- **def-clear path:**
    - caminho que começa onde um nó onde uma variável é definida e que termina onde a mesma é utilizado, sem ser ser redefinida

## Função 1- Path.filename:

## Descrição:

Este método tem como objetivo filtrar o caminho para um ficheiro, de modo a retornar só o nome do mesmo.
Escolhemos esta função pois recebe um parâmetro String que será filtrado e que será usado para criar o nome de um ficheiro.

## Diagrama:


Start
26
def ={path}
c-use = {}
p-use = {}
fileName(path)
35
def ={}
c-use = {}
p-use = {}
Strings.isBlank(path)
false
27
def ={}
c-use = {}
p-use = {path}
!Strings.isBlank(path)
return ""
!Strings.isBlank(path)
true
32
def ={ind}
c-use = {path}
p-use = {}
!path.endsWith("/")
false
29
def ={}
c-use = {}
p-use = {path}
path.endsWith("/")
ind =
path.lastIndexOf('/')
End
path.endsWith("/")
true
30
def ={path}
c-use = {path}
p-use = {}
path =
path.substring(0,
path.length() - 1)
return
path.substring(0, ind
+ 1)
def ={}
c-use = {path, ind}
p-use = {}
33

## Tabelas:

### def use pairs

| def use pairs | | | |
|---|---|---|---|
| variable | path | | |
| pair id | def | use | path |
| 1 | 26 | (27,F) | {26,27,35} |
| 2 | 26 | (27,T) | {26,27,29} |
| 3 | 26 | (29,T) | {26,27,29,30} |
| 4 | 26 | (29,F) | {26,27,29,32} |
| 5 | 30 | 32 | {30,32} |
| 6 | 30 | 33 | {30,32,33} |

| def use pairs | | | |
|---|---|---|---|
| variable | ind | | |
| pair id | def | use | path |
| 1 | 32 | 33 | {32,33} |

### all-defs

| all-defs | | | |
|---|---|---|---|
| variable | path | | |
| pair id | def | use | path |
| 1 | 26 | (27,F) | {26,27,35} |
| 2 | 26 | (27,T) | {26,27,29} |
| 3 | 26 | (29,T) | {26,27,29,30} |
| 4 | 26 | (29,F) | {26,27,29,32} |
| 5 | 30 | 32 | {30,32} |
| 6 | 30 | 33 | {30,32,33} |

| all-defs | | | |
|---|---|---|---|
| variable | ind | | |
| pair id | def | use | path |
| 1 | 32 | 33 | {32,33} |

## all-c-uses

| all-c-uses | | | |
|---|---|---|---|
| variable | path | | |
| pair id | def | use | path |
| 1 | 26 | 30 | {26,27,29,30} |
| 2 | 26 | 33 | {26,27,29,32} |
| 3 | 30 | 32 | {30,32} |
| 4 | 30 | 33 | {30,32,33} |

| all-c-uses | | | |
|---|---|---|---|
| variable | ind | | |
| pair id | def | use | path |
| 1 | 32 | 33 | {32,33} |

## all-p-uses

| all-p-uses | | | |
|---|---|---|---|
| variable | path | | |
| pair id | def | use | path |
| 1 | 26 | (27,F) | {26,27,35} |
| 2 | 26 | (27,T) | {26,27,29} |

| all-p-uses | | | |
|---|---|---|---|
| variable | ind | | |
| pair id | def | use | path |

## all-uses

| all-uses | | | |
|---|---|---|---|
| variable | path | | |
| pair id | def | use | path |
| 1 | 26 | (27,F) | {26,27,35} |
| 2 | 26 | (27,T) | {26,27,29} |
| 3 | 26 | (29,T) | {26,27,29,30} |
| 4 | 26 | (29,F) | {26,27,29,32} |
| 5 | 30 | 32 | {30,32} |
| 6 | 30 | 33 | {30,32,33} |

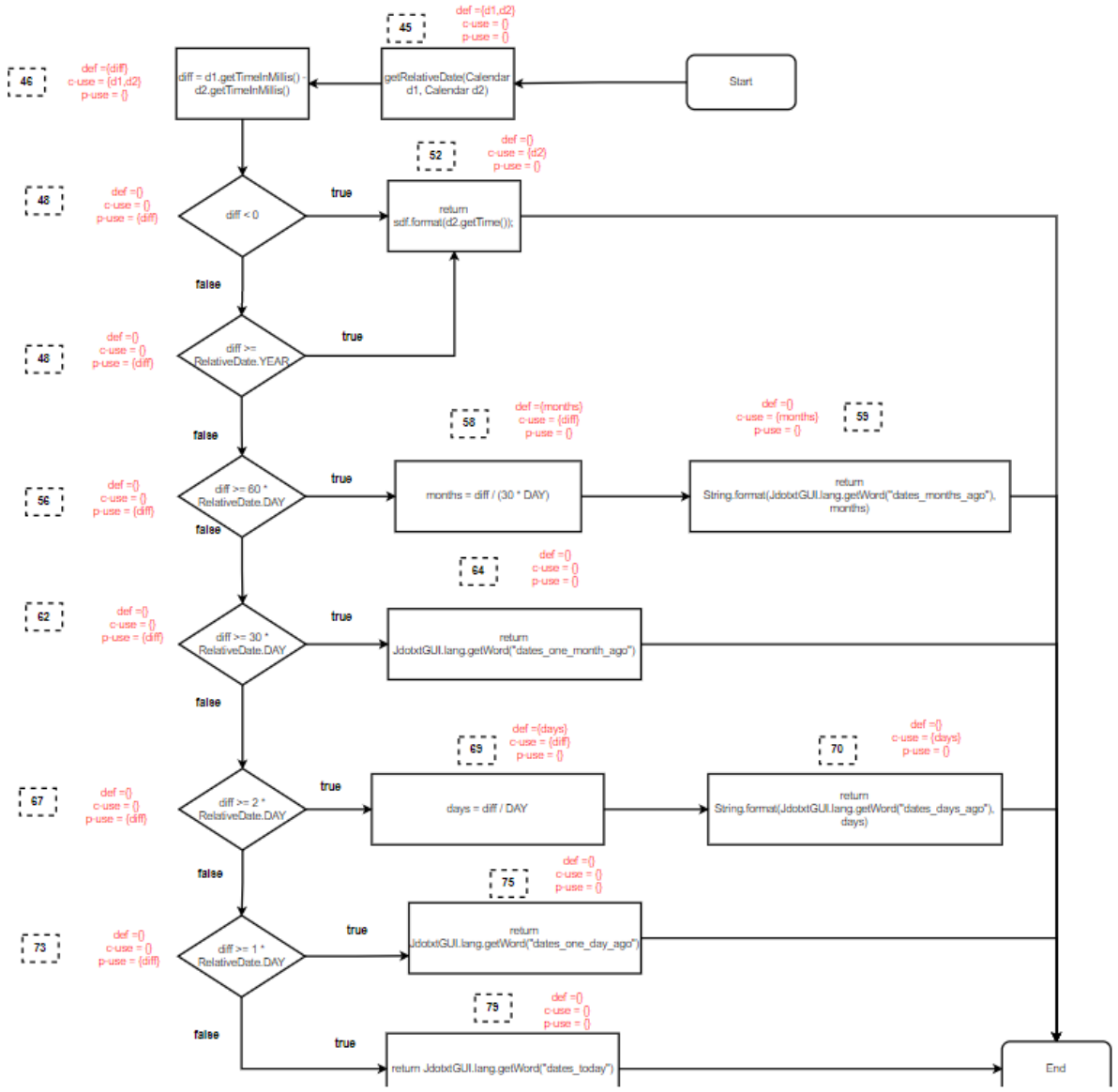
**Análise:**
A partir dos resultados obtidos das tabelas, pode-se perceber como é que as variáveis das função são acedidas, modificadas e, possivelmente, corrompidas.
Realizámos testes ao software com o objetivo de verificar e validar a transferência de dados e, se ocorre, como idealizado no programa.
O método a validar já tinha sido previamente testado, para a técnica de category partition e, por isso, os testes foram reutilizados. Assim, deu para comprovar que os testes feitos anteriormente já cobriam os caminhos possíveis para a tabela *def-use-pairs* da variável path, definida no início do método.

## Função 2- RelativeDate.getRelativeDate:

## Descrição:

O método getRelativeDate devolve uma String com a data relativa, comparando a data dada como parâmetro ao dia de hoje.
Devido à particularidade do tipo do parâmetro e às comparações feitas entre o mesmo e a data atual, decidimos testar as características deste tipo de parâmetro e como é feita a transferência de dados nas variáveis do método.

## Diagrama:


45
def ={d1,d2}
c-use = {}
p-use = {}
getRelativeDate(Calendar d1, Calendar d2)
Start
46
def ={diff}
c-use = {d1,d2}
p-use = {}
diff = d1.getTimeInMillis() - d2.getTimeInMillis()
48
def ={}
c-use = {}
p-use = {diff}
diff < 0
true
52
def ={}
c-use = {d2}
p-use = {}
return sdf.format(d2.getTime());
false
48
def ={}
c-use = {}
p-use = {diff}
diff >= RelativeDate.YEAR
true
false
56
def ={}
c-use = {}
p-use = {diff}
diff >= 60 * RelativeDate.DAY
true
58
def ={months}
c-use = {diff}
p-use = {}
months = diff / (30 * DAY)
def ={}
c-use = {months}
p-use = {}
59
return String.format(JdobxtGUI.lang.getWord("dates_months_ago"), months)
false
64
def ={}
c-use = {}
p-use = {}
62
def ={}
c-use = {}
p-use = {diff}
diff >= 30 * RelativeDate.DAY
true
return JdobxtGUI.lang.getWord("dates_one_month_ago")
false
67
def ={}
c-use = {}
p-use = {diff}
diff >= 2 * RelativeDate.DAY
true
69
def ={days}
c-use = {diff}
p-use = {}
days = diff / DAY
70
def ={}
c-use = {days}
p-use = {}
return String.format(JdobxtGUI.lang.getWord("dates_days_ago"), days)
false
75
def ={}
c-use = {}
p-use = {}
73
def ={}
c-use = {}
p-use = {diff}
diff >= 1 * RelativeDate.DAY
true
return JdobxtGUI.lang.getWord("dates_one_day_ago")
79
def ={}
c-use = {}
p-use = {}
false
true
return JdobxtGUI.lang.getWord("dates_today")
End

## Tabelas:

### def use pairs

| def use pairs | | | |
|---|---|---|---|
| variable | d1 | | |
| pair id | def | use | path |
| 1 | 45 | 46 | {45,46} |

| def use pairs | | | |
|---|---|---|---|
| variable | d2 | | |
| pair id | def | use | path |
| 1 | 45 | 46 | {45,46} |
| 2 | 45 | 52 | {45,46,48,52} |

| def use pairs | | | |
|---|---|---|---|
| variable | diff | | |
| pair id | def | use | path |
| 1 | 46 | 52 | {46,48} |
| 2 | 46 | (48,T) | {46,48,52} |
| 3 | 46 | (48,F) | {46,48,56} |
| 4 | 46 | (56,T) | {46,48,56,58,59} |
| 5 | 46 | (56,F) | {46,48,56,62} |
| 6 | 46 | (62,T) | {46,48,56,62,64} |
| 7 | 46 | (62,F) | {46,48,56,62,67} |
| 8 | 46 | (67,T) | {46,48,56,62,67,69,70} |
| 9 | 46 | (67,F) | {46,48,56,62,67,73} |
| 10 | 46 | (73,T) | {46,48,56,62,67,73,75} |
| 11 | 46 | (73,F) | {46,48,56,62,67,73,79} |

| def use pairs | | | |
|---|---|---|---|
| variable | months | | |
| pair id | def | use | path |
| 1 | 58 | 59 | {58,59} |

| def use pairs | | | |
|---|---|---|---|
| variable | days | | |
| pair id | def | use | path |
| 1 | 69 | 70 | {69,70} |

## all-defs

| all-defs | | | |
|---|---|---|---|
| variable | d1 | | |
| pair id | def | use | path |
| 1 | 45 | 46 | {45,46} |

| all-defs | | | |
|---|---|---|---|
| variable | d2 | | |
| pair id | def | use | path |
| 1 | 45 | 46 | {45,46} |
| 2 | 45 | 52 | {45,46,48,52} |

| all-defs | | | |
|---|---|---|---|
| variable | diff | | |
| pair id | def | use | path |
| 1 | 46 | 52 | {46,48} |
| 2 | 46 | (48,T) | {46,48,52} |
| 3 | 46 | (48,F) | {46,48,56} |
| 4 | 46 | (56,T) | {46,48,56,58,59} |
| 5 | 46 | (56,F) | {46,48,56,62} |
| 6 | 46 | (62,T) | {46,48,56,62,64} |
| 7 | 46 | (62,F) | {46,48,56,62,67} |
| 8 | 46 | (67,T) | {46,48,56,62,67,69,70} |
| 9 | 46 | (67,F) | {46,48,56,62,67,73} |
| 10 | 46 | (73,T) | {46,48,56,62,67,73,75} |
| 11 | 46 | (73,F) | {46,48,56,62,67,73,79} |

| all-defs | | | |
|---|---|---|---|
| variable | months | | |
| pair id | def | use | path |
| 1 | 58 | 59 | {58,59} |

| all-defs | | | |
|---|---|---|---|
| variable | days | | |
| pair id | def | use | path |
| 1 | 69 | 70 | {69,70} |

## all-c-uses

| all-c-uses | | | |
|---|---|---|---|
| variable | d1 | | |
| pair id | def | use | path |
| 1 | 45 | 46 | {45,46} |

| all-c-uses | | | |
|---|---|---|---|
| variable | d2 | | |
| pair id | def | use | path |
| 1 | 45 | 46 | {45,46} |
| 2 | 45 | 52 | {45,46,48,52} |

| all-c-uses | | | |
|---|---|---|---|
| variable | diff | | |
| pair id | def | use | path |
| 1 | 46 | (56,T) | {46,48,56,58,59} |
| 2 | 46 | (67,T) | {46,48,56,62,67,69,70} |

| all-c-uses | | | |
|---|---|---|---|
| variable | months | | |
| pair id | def | use | path |
| 1 | 58 | 59 | {58,59} |

| all-c-uses | | | |
|---|---|---|---|
| variable | days | | |
| pair id | def | use | path |
| 1 | 69 | 70 | {69,70} |

## all-p-uses

| all-p-uses | | | |
|---|---|---|---|
| variable | d1 | | |
| pair id | def | use | path |

| all-p-uses | | | |
|---|---|---|---|
| variable | d2 | | |
| pair id | def | use | path |

| all-p-uses | | | |
|---|---|---|---|
| variable | diff | | |
| pair id | def | use | path |
| 1 | 46 | 52 | {46,48} |
| 2 | 46 | (48,T) | {46,48,52} |
| 3 | 46 | (48,F) | {46,48,56} |
| 4 | 46 | (56,T) | {46,48,56,58,59} |
| 5 | 46 | (56,F) | {46,48,56,62} |
| 6 | 46 | (62,T) | {46,48,56,62,64} |
| 7 | 46 | (62,F) | {46,48,56,62,67} |
| 8 | 46 | (67,T) | {46,48,56,62,67,69,70} |
| 9 | 46 | (67,F) | {46,48,56,62,67,73} |
| 10 | 46 | (73,T) | {46,48,56,62,67,73,75} |
| 11 | 46 | (73,F) | {46,48,56,62,67,73,79} |

| all-p-uses | | | |
|---|---|---|---|
| variable | months | | |
| pair id | def | use | path |

| all-p-uses | | | |
|---|---|---|---|
| variable | days | | |
| pair id | def | use | path |

## all-uses

| all-uses | | | |
|---|---|---|---|
| variable | d1 | | |
| pair id | def | use | path |
| 1 | 45 | 46 | {45,46} |

| all-uses | | | |
|---|---|---|---|
| variable | d2 | | |
| pair id | def | use | path |
| 1 | 45 | 46 | {45,46} |
| 2 | 45 | 52 | {45,46,48,52} |

| all-uses | | | |
|---|---|---|---|
| variable | diff | | |
| pair id | def | use | path |
| 1 | 46 | 52 | {46,48} |
| 2 | 46 | (48,T) | {46,48,52} |
| 3 | 46 | (48,F) | {46,48,56} |
| 4 | 46 | (56,T) | {46,48,56,58,59} |
| 5 | 46 | (56,F) | {46,48,56,62} |
| 6 | 46 | (62,T) | {46,48,56,62,64} |
| 7 | 46 | (62,F) | {46,48,56,62,67} |
| 8 | 46 | (67,T) | {46,48,56,62,67,69,70} |
| 9 | 46 | (67,F) | {46,48,56,62,67,73} |
| 10 | 46 | (73,T) | {46,48,56,62,67,73,75} |
| 11 | 46 | (73,F) | {46,48,56,62,67,73,79} |

| all-uses | | | |
|---|---|---|---|
| variable | months | | |
| pair id | def | use | path |
| 1 | 58 | 59 | {58,59} |

| all-uses | | | |
|---|---|---|---|
| variable | days | | |
| pair id | def | use | path |
| 1 | 69 | 70 | {69,70} |

## Análise:

Os testes de software feitos na entrega *category-partition* vieram a comprovar o que se pode observar nas tabelas, tendo em conta que os testes fazem todos os caminhos identificados no diagrama e todas as possíveis mudanças de dados são verificados pelos testes que os validam.

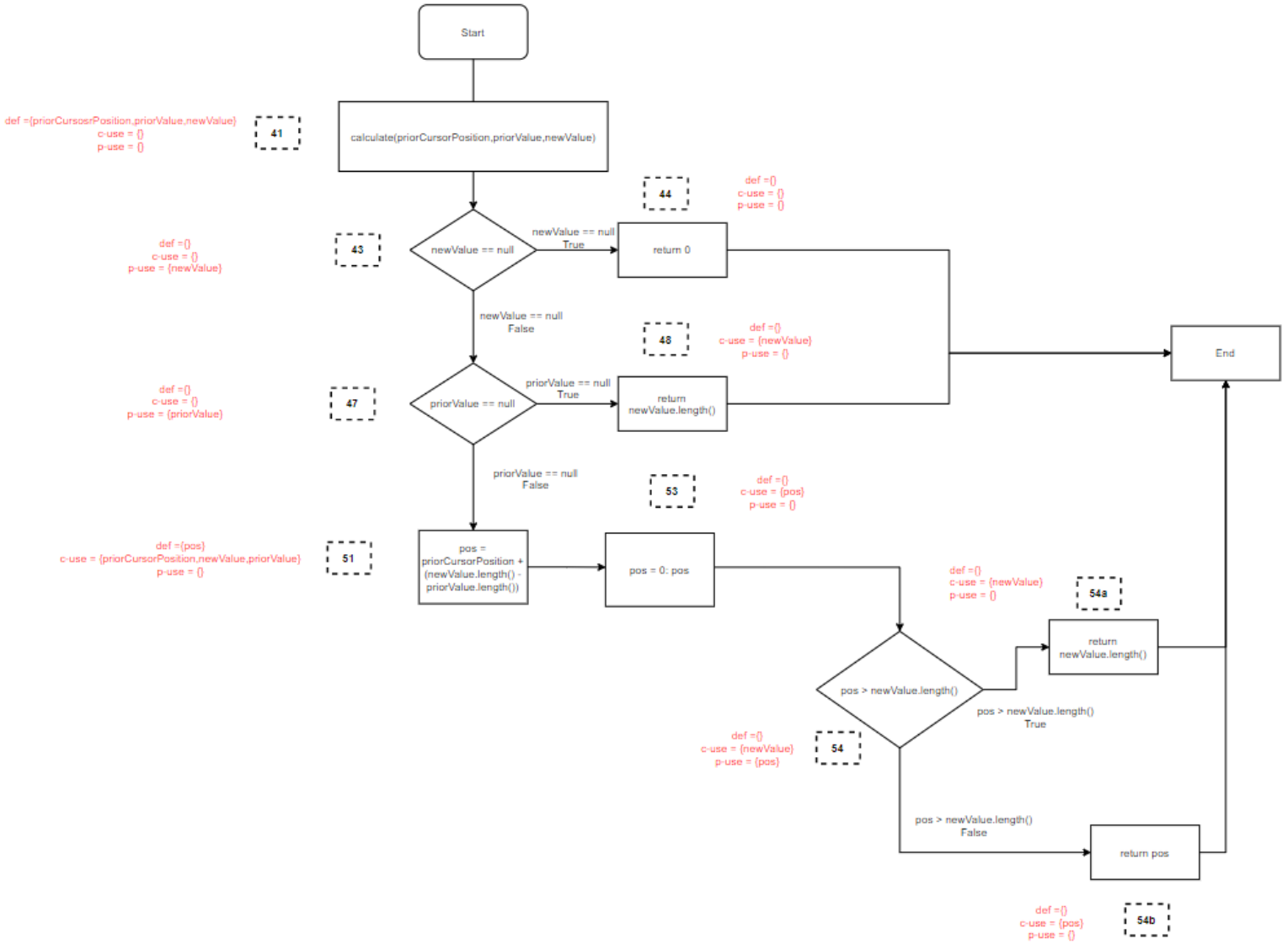
## Função 3- CursorPositionCalculator.calculate:

## Descrição:

Este método tem como objetivo calcular a nova posição do cursor após uma mudança do texto, recebe como parâmetros a posição anterior do cursor, o texto antigo, e o novo texto.

Escolhemos esta função devido à possibilidade de valores null criarem diversos caminhos, e devido à possibilidade de os mesmos serem testados.

## Diagrama:


Start
def ={priorCursosrPosition,priorValue,newValue}
c-use = {}
p-use = {}
41
calculate(priorCursorPosition,priorValue,newValue)
44
def ={}
c-use = {}
p-use = {}
def ={}
c-use = {}
p-use = {newValue}
43
newValue == null
newValue == null
True
return 0
newValue == null
False
48
def ={}
c-use = {newValue}
p-use = {}
End
def ={}
c-use = {}
p-use = {priorValue}
47
priorValue == null
priorValue == null
True
return
newValue.length()
priorValue == null
False
53
def ={}
c-use = {pos}
p-use = {}
def ={pos}
c-use = {priorCursorPosition,newValue,priorValue}
p-use = {}
51
pos =
priorCursorPosition +
(newValue.length() -
priorValue.length())
pos = 0; pos
def ={}
c-use = {newValue}
p-use = {}
54a
return
newValue.length()
pos > newValue.length()
pos > newValue.length()
True
def ={}
c-use = {newValue}
p-use = {pos}
54
pos > newValue.length()
False
return pos
def ={}
c-use = {pos}
p-use = {}
54b

## Tabelas:

### def use pairs

| def use pairs | | | |
|---|---|---|---|
| variable | newValue | | |
| pair id | def | use | path |
| 1 | 41 | (43,T) | {41,43,44} |
| 2 | 41 | (43,F) | {41,43,47} |
| 3 | 41 | 51 | {41,43,47,51} |
| 4 | 41 | 48 | {41,43,47,48} |
| 5 | 41 | (54,T) | {41,43,47,51,53,54,54a} |
| 6 | 41 | (54,F) | {41,43,47,51,53,54,54b} |

| def use pairs | | | |
|---|---|---|---|
| variable | priorValue | | |
| pair id | def | use | path |
| 1 | 41 | (47,T) | {41,43,47,48} |
| 2 | 41 | (47,F) | {41,43,47,51} |
| 3 | 41 | 51 | {41,43,47,51} |

| def use pairs | | | |
|---|---|---|---|
| variable | priorCursorPosition | | |
| pair id | def | use | path |
| 1 | 41 | 51 | {41,43,47,51} |

| def use pairs | | | |
|---|---|---|---|
| variable | pos | | |
| pair id | def | use | path |
| 1 | 51 | 53 | [51,53} |
| 2 | 53 | (54,T) | {51,53,54,54a} |
| 3 | 53 | (54,F) | {51,53,54,54b} |
| 3 | 53 | 54b | {51,53,54,54b} |

## all-defs

| all-defs | | | |
|---|---|---|---|
| variable | newValue | | |
| pair id | def | use | path |
| 1 | 41 | (43,T) | {41,43,44} |
| 2 | 41 | (43,F) | {41,43,47} |
| 3 | 41 | 51 | {41,43,47,51} |
| 4 | 41 | 48 | {41,43,47,48} |
| 1 | 41 | (43,T) | {41,43,44} |
| 2 | 41 | (43,F) | {41,43,47} |

| all-defs | | | |
|---|---|---|---|
| variable | priorValue | | |
| pair id | def | use | path |
| 1 | 41 | (47,T) | {41,43,47,48} |
| 2 | 41 | (47,F) | {41,43,47,51} |
| 3 | 41 | 51 | {41,43,47,51} |

| all-defs | | | |
|---|---|---|---|
| variable | priorCursorPosition | | |
| pair id | def | use | path |
| 1 | 41 | 51 | {41,43,47,51} |

| all-defs | | | |
|---|---|---|---|
| variable | pos | | |
| pair id | def | use | path |
| 1 | 51 | 53 | [51,53} |
| 2 | 53 | (54,T) | {51,53,54,54a} |
| 3 | 53 | (54,F) | {51,53,54,54b} |
| 3 | 53 | 54b | {51,53,54,54b} |

## all-c-uses

| all-c-uses | | | |
|---|---|---|---|
| variable | newValue | | |
| pair id | def | use | path |
| 1 | 41 | 51 | {41,43,47,51} |
| 2 | 41 | 48 | {41,43,47,48} |
| 3 | 41 | 54 | {41,43,47,48,53,54} |
| 4 | 41 | 54a | {41,43,47,48,53,54,54a} |

| all-c-uses | | | |
|---|---|---|---|
| variable | priorValue | | |
| pair id | def | use | path |
| 1 | 41 | 51 | {41,43,47,51} |

| all-c-uses | | | |
|---|---|---|---|
| variable | priorCursorPosition | | |
| pair id | def | use | path |
| 1 | 41 | 51 | {41,43,47,51} |

| all-c-uses | | | |
|---|---|---|---|
| variable | priorCursorPosition | | |
| pair id | def | use | path |
| 1 | 51 | 53 | [51,53} |
| 2 | 53 | 54b | {51,53,54,54b} |

## all-p-uses

| all-p-uses | | | |
|---|---|---|---|
| variable | newValue | | |
| pair id | def | use | path |
| 1 | 41 | (43,T) | {41,43,44} |
| 2 | 41 | (43,F) | {41,43,47} |
| 3 | 41 | (43,T) | {41,43,44} |
| 4 | 41 | (43,F) | {41,43,47} |

| all-p-uses | | | |
|---|---|---|---|
| variable | priorValue | | |
| pair id | def | use | path |
| 41 | (47,T) | {41,43,47,48} | {41,43,47,51} |
| 41 | (47,F) | {41,43,47,51} | {41,43,47,51} |

| all-p-uses | | | |
|---|---|---|---|
| variable | priorCursorPosition | | |
| pair id | def | use | path |
| 1 | 53 | (54,T) | {51,53,54,54a} |
| 2 | 53 | (54,F) | {51,53,54,54b} |

## all-uses

| all-uses | | | |
|---|---|---|---|
| variable | newValue | | |
| pair id | def | use | path |
| 1 | 41 | (43,T) | {41,43,44} |
| 2 | 41 | (43,F) | {41,43,47} |
| 3 | 41 | 51 | {41,43,47,51} |
| 4 | 41 | 48 | {41,43,47,48} |
| 5 | 41 | (43,T) | {41,43,44} |
| 6 | 41 | (43,F) | {41,43,47} |
| 7 | 41 | 54 | {41,43,47,48,53,54} |
| 8 | 41 | 54a | {41,43,47,48,53,54,54a} |

| all-uses | | | |
|---|---|---|---|
| variable | priorValue | | |
| pair id | def | use | path |
| 1 | 41 | (47,T) | {41,43,47,48} |
| 2 | 41 | (47,F) | {41,43,47,51} |
| 3 | 41 | 51 | {41,43,47,51} |

| all-uses | | | |
|---|---|---|---|
| variable | priorCursorPosition | | |
| pair id | def | use | path |
| 1 | 41 | 51 | {41,43,47,51} |

| def use pairs | | | |
|---|---|---|---|
| variable | pos | | |
| pair id | def | use | path |
| 1 | 51 | 53 | [51,53} |
| 2 | 53 | (54,T) | {51,53,54,54a} |
| 3 | 53 | (54,F) | {51,53,54,54b} |
| 3 | 53 | 54b | {51,53,54,54b} |

## Análise:

Através das tabelas é possível analisar o percurso percorrido pelo código e os caminhos onde as variáveis são utilizadas, tanto para leitura como para verificação condicional.
Estes caminhos já tinham sido testados anteriormente quando foi feito o relatório de *category-partition*, e os testes comprovaram que os caminhos descritos pelas tabelas são de facto corretos, e que já estavam cobertos.