Swen2 Jeroen Vermaat Individuele opdracht

# Grenswaarde analyse:

Package: Controller

Class: GameAcceptionThreadController

Methode: run()

Regel: 42

|  |  |  |  |
| --- | --- | --- | --- |
|  | 0 | 1 | 2 |
| AmountRefused >= 1 | 0 | 1 | 1 |
| AmountRefused > 1 | 0 | 0 | 1 |
| AmountRefused == 1 | 0 | 1 | 0 |
| AmountRefused != 1 | 1 | 0 | 1 |
| AmountRefused <= 1 | 1 | 1 | 0 |
| AmountRefused < 1 | 1 | 0 | 0 |
| Lichte variant |  | X | X |
| Normale variant | X | X | X |

# Pairwise testing:

Package: Controller

Class: ToolCardController

Methode: isValidEglomiseBrushPlacement()

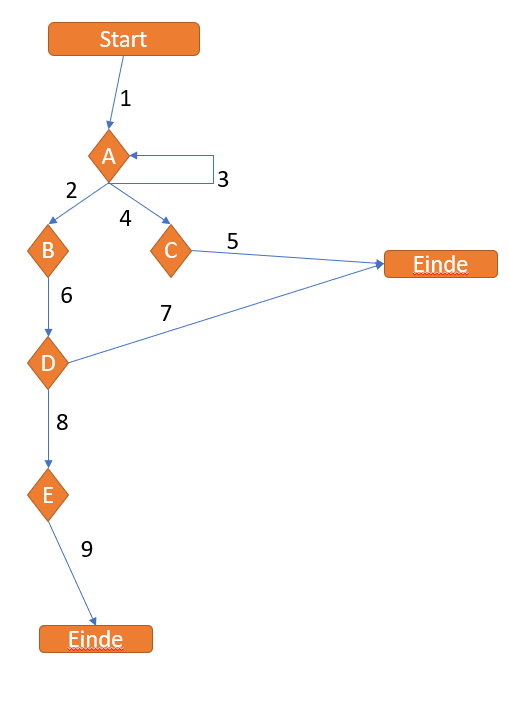
Regel: 371

Stukje code: if (chosenCard.hasDoubleSurroundingColor(receiverX, receiverY, dice.getColor()) && chosenCard.hasSurroundingDice(receiverX, receiverY)) {

|  |  |  |
| --- | --- | --- |
| A | B | A && B |
| 1 | 1 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 0 |
| 0 | 0 | 0 |

Uit deze table is af te lezen dat de vergelijking tussen de variabelen alleen goed is als beide variabelen “True” zijn. Elk ander geval betekend dat dat stukje code niet verder mag.

Procescyclus testing:



|  |  |  |  |
| --- | --- | --- | --- |
| Knooppunt | Inkomend | Uitgaand | Combinaties |
| A | 1, 3 | 2, 3, 4 | 1-2, 1-4, 1-3, 3-2, 3-3. 3-4 |
| B | 2 | 6 | 2-6 |
| C | 4 | 5 | 4-5 |
| D | 6 | 8 | 6-8 |
| E | 8 | 9 | 8-9 |

* Testgeval 1: 1, 2, 6, 8, 9
* Testgeval 2: 1, 3, 2, 6, 8, 9
* Testgeval 3: 1, 4, 5
* Testgeval 4: 1, 3, 4, 5
* Testgeval 5: 1, 2, 6, 7
* Testgeval 5: 1, 3, 2, 6, 7

Package: Controller

Class: GameAcceptionThreadController

Functie: Wat er allemaal moet gebeuren om een invitation te accepteren.