|  |  |
| --- | --- |
| **ID** | 9 |
| **Verfahren** | Parametrisierter JUnit-Test |
| **Klasse** | Colony |
| **Methoden** | updatePheromone(City a, City b, double pLevel)  Adds the given value to the value in the matrix between a and b |
| **Vorbedingung** | Pheromonmatrix ist mit 1 normalisiert |
| **Eingaben** | City(1), City(2), 5 |
| **Erwartetes Ergebnis** | Pheromon zwischen a und b == 6 |
| **Ergebnis** | Pheromon zwischen a und b == 6  **Test erfolgreich** |

|  |  |
| --- | --- |
| **ID** | 10 |
| **Verfahren** | Parametrisierter JUnit-Test |
| **Klasse** | Colony |
| **Methoden** | initPheromone()  Die Pheromonmatrix muss in der Größe der Distanzmatrix von Landscape mit je 1 als Pheromonwert initialisiert werden |
| **Vorbedingung** | City(1), City(2) |
| **Eingaben** | - |
| **Erwartetes Ergebnis** | Größe der Pheromonmatrix == 2  Kolonie erkennt Pheromonmatrix als initialisiert  Pheromonwert zwischen City(1) und City(2) == 1 |
| **Ergebnis** | Größe der Pheromonmatrix == 2  Kolonie erkennt Pheromonmatrix als initialisiert  Pheromonwert zwischen City(1) und City(2) == 1  **Test erfolgreich** |

|  |  |
| --- | --- |
| **ID** | 11 |
| **Verfahren** | Parametrisierter JUnit-Test |
| **Klasse** | Colony |
| **Methoden** | initPheromone()  Die Pheromonmatrix muss in der Größe der Distanzmatrix von Landscape mit je 1 als Pheromonwert initialisiert werden |
| **Vorbedingung** | - |
| **Eingaben** | - |
| **Erwartetes Ergebnis** | PheromoneInitializationException wird geworfen |
| **Ergebnis** | PheromoneInitializationException wird geworfen  **Test erfolgreich** |

|  |  |
| --- | --- |
| **ID** | 12 |
| **Verfahren** | Parametrisierter JUnit-Test |
| **Klasse** | Colony |
| **Methoden** | initAnts()  Die Threads müssen zu Beginn jeder Generation erstellt werden |
| **Vorbedingung** | - |
| **Eingaben** | - |
| **Erwartetes Ergebnis** | Anzahl der Threads stimmt mit der konfigurierten Anzahl überein |
| **Ergebnis** | Anzahl der Threads stimmt mit der konfigurierten Anzahl überein  **Test erfolgreich** |

|  |  |
| --- | --- |
| **ID** | 13 |
| **Verfahren** | Parametrisierter JUnit-Test |
| **Klasse** | Colony |
| **Methoden** | killAnt(Ant a)  Sollte eine Ameise auf einen schwerwiegenden Fehler stoßen, kann sie sich selbst beenden und der Kolonie mitteilen die Referenz zu löschen |
| **Vorbedingung** | Kolonie hat derzeit eine Liste *ants* von Ameisen |
| **Eingabenµ** | Erste Ameise aus *ants* |
| **Erwartetess Ergebnis** | *ants* enthält nicht mehr die zuvor erste Ameise |
| **Ergebnis** | *ants* enthält nicht mehr die zuvor erste Ameise  **Test erfolgreich** |

|  |  |
| --- | --- |
| **ID** | 14 |
| **Verfahren** | Parametrisierter JUnit-Test |
| **Klasse** | Colony |
| **Methoden** | notifyColony()  Sobald eine Generation beendet ist, wird eine neue erzeugt |
| **Vorbedingung** | City(1), City(2)  (Städte haben untereinander die Distanz 2) |
| **Eingaben** | - |
| **Erwartetes Ergebnis** | Liste der Ameisen der Kolonie ist größer 0  Generationenzähler steht auf 2 |
| **Ergebnis** | Liste der Ameisen der Kolonie ist größer 0  Generationenzähler steht auf 2  **Test erfolgreich** |