

Scenariusz 8

Badanie zachowań

1. Utwórz klasę agenta o nazwie `klasa_1_2`. Agent ten powinien wykonywać zachowanie odwzorowujące następującą maszynę skończenia stanową:

Uwagi: Stany A, C i E polegają na wypisaniu nazwy stanu. Przejścia z tych stanów następują bezwarunkowo dalej. W stanach B i D również następuje wypisanie nazwy stanu, ale oprócz tego losowana jest liczba ze zbioru 0 i 1, która jest zwracana w chwili kończenia się zachowań związanych ze stanami.

```
-----  
Executing behaviour A  
Executing behaviour B  
Exit value is 1  
Executing behaviour C  
Executing behaviour D  
Exit value is 1  
Executing behaviour A  
Executing behaviour B  
Exit value is 1  
Executing behaviour C  
Executing behaviour D  
Exit value is 0  
Executing behaviour E  
FSM behaviour completed.  
  
fsm.registerFirstState(new NamePrinter(), STATE_A);  
fsm.registerState(new RandomGenerator( max: 2), STATE_B);  
fsm.registerState(new NamePrinter(), STATE_C);  
fsm.registerState(new RandomGenerator( max: 2), STATE_D);  
fsm.registerLastState(new NamePrinter(), STATE_E);  
  
fsm.registerDefaultTransition(STATE_A, STATE_B);  
fsm.registerTransition(STATE_B, STATE_C, event: 1);  
fsm.registerDefaultTransition(STATE_C, STATE_D);  
fsm.registerTransition(STATE_B, STATE_D, event: 0);  
fsm.registerTransition(STATE_D, STATE_A, event: 1);  
fsm.registerTransition(STATE_D, STATE_E, event: 0);  
addBehaviour(fsm);
```

2. Przypomnij sobie czwarte polecenie z ostatnich zajęć: Utwórz klasę agenta o nazwie `klasa_4` na podstawie kodu `klasa_1`. Do agenta dodaj zachowanie „generyczne”, polegające na wykonaniu trzech kroków:

- W pierwszym kroku wypisuje „pierwszy krok”,
- W drugim kroku wypisuje „drugi krok”,
- W trzecim kroku wypisuje „trzeci krok” i zachowanie zostaje usunięte z puli zachowań agenta. Znajdź lub napisz odpowiedni kod.

```
-----  
Agent xx started.
```

```
1  
2  
1  
3
```

```
case 1:  
    // Perform operation 1: print out a message  
    myAgent.addBehaviour(new CyclicBehaviour() {  
        @Override  
        public void action() {  
            System.out.println("1");  
        }  
    });  
    break;  
case 2:  
    myAgent.addBehaviour(new OneShotBehaviour(myAgent) {  
        public void action() { System.out.println("2"); }  
    });  
    break;  
case 3:  
    Behaviour b=new OneShotBehaviour() {  
        @Override  
        public void action() {  
            System.out.println("3");  
            removeBehaviour( b: this);  
        }  
    };  
    myAgent.addBehaviour(b);  
    break;
```

3. Utwórz zachowanie, które będzie polegało na równoległym wykonywaniu trzech zachowań „generycznych” z zadania poprzedniego (z pliku klasa_4.java). Zachowanie dodaj do agenta, którego klasę nazwiesz klasa_2_3. Prześledź działanie agenta.

```
// Add the generic behaviour
//addBehaviour(new ThreeStepBehaviour());
ParallelBehaviour p=new ParallelBehaviour();
p.addSubBehaviour(new ThreeStepBehaviour());
p.addSubBehaviour(new ThreeStepBehaviour());
p.addSubBehaviour(new ThreeStepBehaviour());
addBehaviour(p);
}
```

```
Agent xx started.
1
1
1
1
2
1
1
2
1
1
3
1
1
3
```

4. Utwórz zachowanie, które będzie polegało na sekwencyjnym wykonywaniu trzech zachowań „generycznych” z zadania drugiego (z klasa_4.java). Zachowanie dodaj do agenta, którego klasę nazwiesz klasa_2_4. Prześledź działanie agenta.

```
// addBehaviour(p);
SequentialBehaviour s=new SequentialBehaviour();
ThreeStepBehaviour b1=new ThreeStepBehaviour(),b2=new ThreeStepBehav
s.addSubBehaviour(b1);
// s.addSubBehaviour(new OneShot());
s.addSubBehaviour(b2);
addBehaviour(s);
System.out.println("Agent "+getLocalName()+" ended.");
```

Agent xx started.

Agent xx ended.

```
1
2 RUNNING
1
3
1
1
1
1
2 RUNNING
1
3
1
```

5. Utwórz agenta, który będzie wykonywał dwa zachowania cykliczne (wypisujące odpowiednio „cyclic 1” oraz „cyclic 2”) w dwóch osobnych wątkach. Klasa agenta ma się nazywać `klasa_2_5`. Prześledź działanie agenta.

```
system.out.println("Agent  
Cyclic b1=new Cyclic();  
b1.count=1;  
Cyclic b2=new Cyclic();  
b2.count=2;  
addBehaviour(b1);  
addBehaviour(b2);
```

```
Agent xx started.  
cyclic1  
cyclic2  
cyclic1  
cyclic2  
cyclic1  
cyclic2  
cyclic1  
cyclic2  
cyclic1  
cyclic2
```