

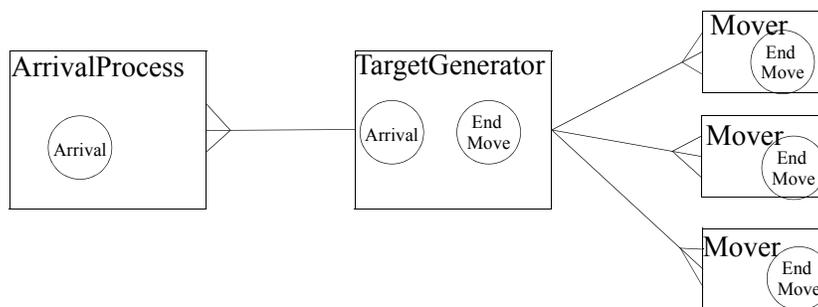
Computer Lab 05

Objectives

- Create a class that generates targets and does the appropriate registering and unregistering of referee and listeners

Description

The targets can be instances of `UniformLinearMover`. They are not created all at once in the beginning of the simulation, but must be created periodically. Therefore, a class that creates the targets at the appropriate times will be written. Call the class `TargetGenerator`. Since the stimulus for creating a new target is an `ArrivalProcess`, it should seem natural to have the instance of `TargetGenerator` listen to an `ArrivalProcess` for the signal to instantiate a new target. Also, bookkeeping will be done when a target has completed its journey, so each target also needs to be listened to for its `EndMove` event.



As each target is created, the `TargetGenerator` will add itself to the target as a `SimEventListener`. When the target has finished its move, it will remove itself as a `SimEventListener` to the target. Since the targets will not be reused, they need to be unregistered from `Schedule`. Therefore, after instantiating a target, invoke `Schedule.removeRerun(target)`, where `target` is the `UniformLinearMover` instance.

For the sensor(s) to have a chance at detection, each target must be registered with an appropriate Referee object. So, upon creation, the target is registered with a Referee, and upon completion of its journey, the target is unregistered from the Referee. The unregistering is important because after its journey, the target will not be detected again. Yet if the targets are not unregistered, they will stay on the Referee's list of potential targets. As the number of iterations increases, the Referee will have more and more work to do.

The `TargetGenerator` class can be made specific to this task. Therefore, declare parameters for the channel width, the speed of the generated targets, the distance the targets will travel, and a `RandomVariate` for the random starting location. Also, define a parameter of type `SensorTargetReferee` that will be the referee instance each target will be registered with when created (and unregistered when finished moving). Each parameter should have its value passed in through the constructor.

Define state variables that count the number of targets generated and the number of targets that have arrived. Also, define a state variable of type `HashSet` that will contain the targets that have been generated in each iteration. When a target is created, add it to this set, and when it reaches its destination, remove it. After each simulation run, there may be targets left. Therefore, the `reset()` method should iterate through the created targets set and unregister each one from the referee.

Test Class

To test your `TargetGenerator` class, write an execution method called `TestTargetGenerator`. In its `main` method, create a stationary `CookieCutterSensor` in the center of the channel (say at (150, 250) with a range of 150. Register it with an instance of `SensorTargetReferee`, and pass that same `SensorTargetReferee` instance to the constructor of your `TargetGenerator` instance. Instantiate your `ArrivalProcess` and have the `TargetGenerator` instance listen to it. Run in single-step mode to ensure that the targets are being created, sent on their way, and are detected (and undetected) by the sensor. Use a Distance of 500 to ensure that the sensor will be “cleared.”

Sample Output

Some sample output is shown below. Your exact numbers may be different, but you should verify that the `Detections` are occurring properly. For the `CookieCutterSensor` this is straightforward. (For the project, of course, your sensor will be a `ConstantRateSensor`.)

```
Arrival Process
  Interarrival Times: Gamma (2.0, 1.5)
TargetGenerator
  Target Speed:    25.0
  Distance:        500.0
  Initial Location:      Uniform (0.0, 300.0)
CookieCutterSensor (150.0) [Stationary (150.000,250.000) [0.000,0.000]]

** Event List -- Starting Simulation **
0.000  Run
0.000  Run
30.000 Stop
** End of Event List -- Starting Simulation **

Time: 0.000      Current Event: Run      [1]
** Event List -- **
0.000  Run
1.574  Arrival
30.000 Stop
** End of Event List -- **

Time: 0.000      Current Event: Run      [2]
** Event List -- **
1.574  Arrival
30.000 Stop
** End of Event List -- **

Time: 1.574      Current Event: Arrival  [1]
** Event List -- **
1.574  StartMove      {simkit.smdx.UniformLinearMover.9 (277.359,0.000) [0.000,25.000]}
5.951  Arrival
30.000 Stop
** End of Event List -- **

Time: 1.574      Current Event: StartMove      {simkit.smdx.UniformLinearMover.9 (277.359,0.000)
[0.000,25.000]} [1]
** Event List -- **
5.951  Arrival
8.404  EnterRange      {CookieCutterSensor (150.0) [Stationary (150.000,250.000) [0.000,0.000]],
simkit.smdx.UniformLinearMover.9 (277.359,0.000) [0.000,25.000]}
21.574 EndMove      {simkit.smdx.UniformLinearMover.9 (277.359,0.000) [0.000,25.000]}
30.000 Stop
** End of Event List -- **

Time: 5.951      Current Event: Arrival  [2]
** Event List -- **
```

```

5.951 StartMove      {simkit.smdx.UniformLinearMover.10 (53.488,0.000) [0.000,25.000]}
8.338 Arrival
8.404 EnterRange    {CookieCutterSensor (150.0) [Stationary (150.000,250.000) [0.000,0.000]],
simkit.smdx.UniformLinearMover.9 (277.359,109.420) [0.000,25.000]}
21.574 EndMove      {simkit.smdx.UniformLinearMover.9 (277.359,109.420) [0.000,25.000]}
30.000 Stop
** End of Event List -- **

Time: 5.951      Current Event: StartMove      {simkit.smdx.UniformLinearMover.10 (53.488,0.000)
[0.000,25.000]}      [2]
** Event List -- **
8.338 Arrival
8.404 EnterRange    {CookieCutterSensor (150.0) [Stationary (150.000,250.000) [0.000,0.000]],
simkit.smdx.UniformLinearMover.9 (277.359,109.420) [0.000,25.000]}
11.358 EnterRange   {CookieCutterSensor (150.0) [Stationary (150.000,250.000) [0.000,0.000]],
simkit.smdx.UniformLinearMover.10 (53.488,0.000) [0.000,25.000]}
21.574 EndMove      {simkit.smdx.UniformLinearMover.9 (277.359,109.420) [0.000,25.000]}
25.951 EndMove      {simkit.smdx.UniformLinearMover.10 (53.488,0.000) [0.000,25.000]}
30.000 Stop
** End of Event List -- **

Time: 8.338      Current Event: Arrival [3]
** Event List -- **
8.338 StartMove      {simkit.smdx.UniformLinearMover.11 (72.825,0.000) [0.000,25.000]}
8.404 EnterRange     {CookieCutterSensor (150.0) [Stationary (150.000,250.000) [0.000,0.000]],
simkit.smdx.UniformLinearMover.9 (277.359,169.106) [0.000,25.000]}
11.000 Arrival
11.358 EnterRange    {CookieCutterSensor (150.0) [Stationary (150.000,250.000) [0.000,0.000]],
simkit.smdx.UniformLinearMover.10 (53.488,59.686) [0.000,25.000]}
21.574 EndMove      {simkit.smdx.UniformLinearMover.9 (277.359,169.106) [0.000,25.000]}
25.951 EndMove      {simkit.smdx.UniformLinearMover.10 (53.488,59.686) [0.000,25.000]}
30.000 Stop
** End of Event List -- **

Time: 8.338      Current Event: StartMove      {simkit.smdx.UniformLinearMover.11 (72.825,0.000)
[0.000,25.000]}      [3]
** Event List -- **
8.404 EnterRange     {CookieCutterSensor (150.0) [Stationary (150.000,250.000) [0.000,0.000]],
simkit.smdx.UniformLinearMover.9 (277.359,169.106) [0.000,25.000]}
11.000 Arrival
11.358 EnterRange    {CookieCutterSensor (150.0) [Stationary (150.000,250.000) [0.000,0.000]],
simkit.smdx.UniformLinearMover.10 (53.488,59.686) [0.000,25.000]}
13.193 EnterRange    {CookieCutterSensor (150.0) [Stationary (150.000,250.000) [0.000,0.000]],
simkit.smdx.UniformLinearMover.11 (72.825,0.000) [0.000,25.000]}
21.574 EndMove      {simkit.smdx.UniformLinearMover.9 (277.359,169.106) [0.000,25.000]}
25.951 EndMove      {simkit.smdx.UniformLinearMover.10 (53.488,59.686) [0.000,25.000]}
28.338 EndMove      {simkit.smdx.UniformLinearMover.11 (72.825,0.000) [0.000,25.000]}
30.000 Stop
** End of Event List -- **

Time: 8.404      Current Event: EnterRange      {CookieCutterSensor (150.0) [Stationary
(150.000,250.000) [0.000,0.000]], simkit.smdx.UniformLinearMover.9 (277.359,170.756)
[0.000,25.000]}      [1]
** Event List -- **
8.404 Detection      {Contact: [277.359, 170.756]}
11.000 Arrival
11.358 EnterRange    {CookieCutterSensor (150.0) [Stationary (150.000,250.000) [0.000,0.000]],
simkit.smdx.UniformLinearMover.10 (53.488,61.336) [0.000,25.000]}
13.193 EnterRange    {CookieCutterSensor (150.0) [Stationary (150.000,250.000) [0.000,0.000]],
simkit.smdx.UniformLinearMover.11 (72.825,1.650) [0.000,25.000]}
14.744 ExitRange     {CookieCutterSensor (150.0) [Stationary (150.000,250.000) [0.000,0.000]],
simkit.smdx.UniformLinearMover.9 (277.359,170.756) [0.000,25.000]}
21.574 EndMove      {simkit.smdx.UniformLinearMover.9 (277.359,170.756) [0.000,25.000]}
25.951 EndMove      {simkit.smdx.UniformLinearMover.10 (53.488,61.336) [0.000,25.000]}
28.338 EndMove      {simkit.smdx.UniformLinearMover.11 (72.825,1.650) [0.000,25.000]}

```

```
30.000 Stop
** End of Event List -- **

Time: 8.404      Current Event: Detection      {Contact: [277.359, 170.756]} [1]
** Event List -- **
11.000 Arrival
11.358 EnterRange      {CookieCutterSensor (150.0) [Stationary (150.000,250.000) [0.000,0.000]],
simkit.smdx.UniformLinearMover.10 (53.488,61.336) [0.000,25.000]}
13.193 EnterRange      {CookieCutterSensor (150.0) [Stationary (150.000,250.000) [0.000,0.000]],
simkit.smdx.UniformLinearMover.11 (72.825,1.650) [0.000,25.000]}
14.744 ExitRange       {CookieCutterSensor (150.0) [Stationary (150.000,250.000) [0.000,0.000]],
simkit.smdx.UniformLinearMover.9 (277.359,170.756) [0.000,25.000]}
21.574 EndMove         {simkit.smdx.UniformLinearMover.9 (277.359,170.756) [0.000,25.000]}
25.951 EndMove         {simkit.smdx.UniformLinearMover.10 (53.488,61.336) [0.000,25.000]}
28.338 EndMove         {simkit.smdx.UniformLinearMover.11 (72.825,1.650) [0.000,25.000]}
30.000 Stop
** End of Event List -- **
```

Deliverables

There are no deliverables for this lab. The class you build should be useful in implementing Project 2.