Lesson #11: Elementary Queueing Theory - Birth Death Simulations

■ Simulating Markov Chains

```
rRow[distlist_List] := Module[{},

  (*This function should return a random variable with distribution as distList*)
  RandomInteger[{1, Length[distlist]}]
 ]
```

In[6]:=
```
rRow[distlist_List] := RandomChoice[distlist → Table[i, {i, Length[distlist]}]]
```

In[10]:=
```
pp = {{0.2`, 0.1`, 0.7`}, {0.499`, 0.001`, 0.5`}, {0.001`, 0.199`, 0.8`}};
pp // MatrixForm
```

Out[11]//MatrixForm=

$$\begin{pmatrix} 0.2 & 0.1 & 0.7 \\ 0.499 & 0.001 & 0.5 \\ 0.001 & 0.199 & 0.8 \end{pmatrix}$$

In[12]:=
```
next[i_] := rRow[pp[[i]]]
NestList[next, 2, 100]
```

Out[13]=
```
{2, 1, 1, 1, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 2, 1, 3, 3, 3, 3, 3, 2, 3,
 3, 3, 3, 2, 3, 3, 3, 3, 3, 2, 1, 3, 3, 3, 3, 3, 3, 2, 3, 3, 3, 3, 3, 3, 3, 3,
 2, 1, 3, 3, 3, 2, 3, 3, 3, 3, 3, 2, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 2, 3, 3,
 3, 3, 2, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 2, 3, 3, 3}
```

In[14]:=
```
pp = {{0.2, 0.1, 0.7},
   {0.5, 0.01, 0.49},
   {0.01, 0.19, 0.8}};
rRow[distlist_List] := RandomChoice[distlist → Table[i, {i, Length[distlist]}]];
NestList[rRow[pp[[#]]] &, 2, 100]
```

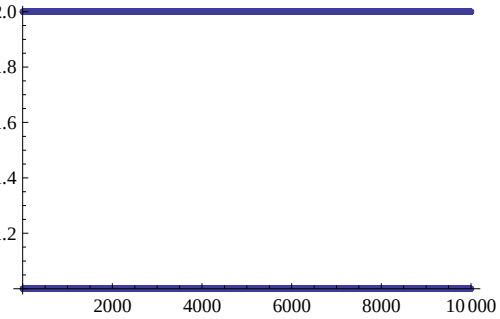
Out[16]=
```
{2, 1, 3, 2, 3, 2, 3, 3, 2, 1, 3, 3, 3, 3, 3, 3, 3, 2, 3, 3, 3, 3,
 3, 3, 3, 3, 2, 1, 3, 3, 3, 2, 1, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3,
 2, 3, 3, 3, 2, 1, 2, 3, 3, 2, 1, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 2,
 1, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 2, 1, 3, 2, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3}
```

In[18]:=
```
pp = {{0.2, 0.8},
   {0.5, 0.5}};
rRow[distlist_List] := RandomChoice[distlist → Table[i, {i, Length[distlist]}]];
rel = NestList[rRow[pp[[#]]] &, 1, 10 000];
Short[rel]
```

Out[21]//Short=
```
{1, 2, 1, 2, 2, 2, 1, 2, 1, 1, 1, 1, 1, 2, <<9973>>, 2, 2, 2, 2, 1, 2, 1, 2, 2, 1, 2, 2, 1, 2}
```

In[24]:=
```
ListPlot[rel]
```

Out[24]=



■ Simulating Markov Jump Processes

Discussion Only

■ Simulating a Birth Death Process - Continuation in next class

■ Working on a Queueing Demonstrations