How long will you wait?

The problem

Suppose you run a customer service center and you know that on average 8 people an hour will come in to be helped. Is it better to have 1 representative who can help 10 people an hour on average or 2 people who can help 5 an hour on average?

We can model this in *Mathematica* with a command called QueueingProcess.

QueueingProcess[*m*, *n*, *k*] represents a queue process where *m* people walk in per hour on average and they can be helped by *k* people who can each help *n* people per hour.

So the first type of queue is represented by QueueingProcess[8, 10, 1]:

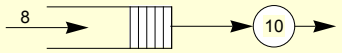
```
queue1 = QueueingProcess[8, 10, 1];
```

The second type of queue is represented by Queueing Process[8,5,2]

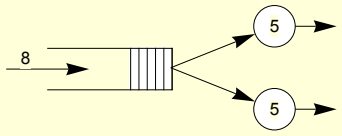
```
queue2 = QueueingProcess[ 8, 5, 2];
```

We can compare these side-by-side:

```
Grid[{{QueueProperties[queue1], QueueProperties[queue2]}}, Alignment → Top]
```



Basic Properties	
QueueNotation	M/M/1
ArrivalRate	8
ServiceRate	10
UtilizationFactor	$\frac{4}{5}$
Throughput	8
ServiceChannels	1
SystemCapacity	∞
InitialState	0
Performance Measures	
MeanSystemSize	4
MeanSystemTime	$\frac{1}{2}$
MeanQueueSize	$\frac{16}{5}$
MeanQueueTime	$\frac{2}{5}$



Basic Properties	
QueueNotation	M/M/2
ArrivalRate	8
ServiceRate	5
UtilizationFactor	$\frac{4}{5}$
Throughput	8
ServiceChannels	2
SystemCapacity	∞
InitialState	0
Performance Measures	
MeanSystemSize	$\frac{40}{9}$
MeanSystemTime	$\frac{5}{9}$
MeanQueueSize	$\frac{128}{45}$
MeanQueueTime	$\frac{16}{45}$

By comparing the MeanSystemTime for each case we can see which case handles people more quickly:

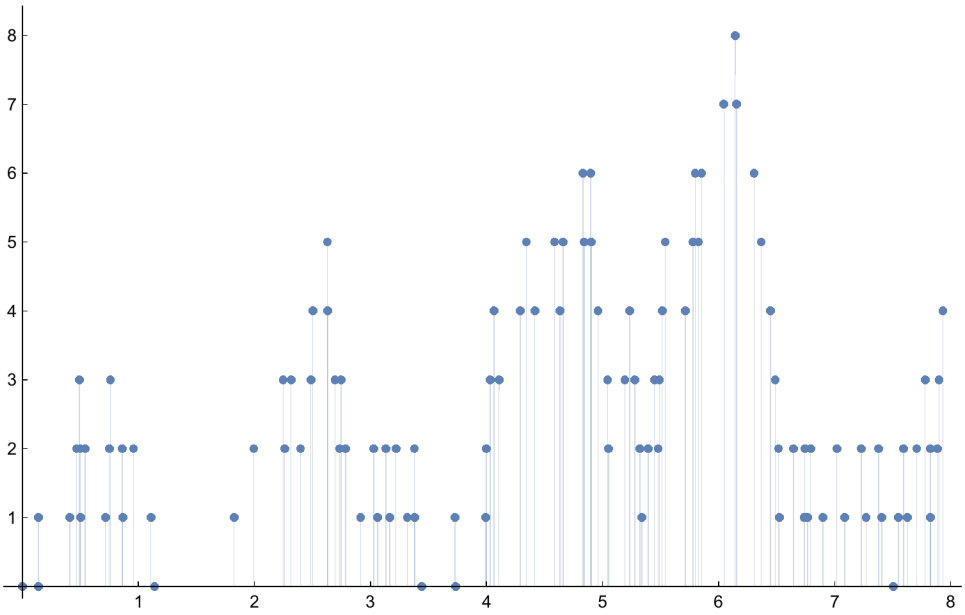
```
N[{1/2, 5/9}]
```

```
{0.5, 0.555556}
```

It turns out the quicker average service for the single-representative system more than outweighs the extra time spent in line

Here is a simulation of the number of people in the system for the 2 representative queue:

```
ListPlot[ RandomFunction[ queue2, {0, 8}], Filling → Axis,
 Ticks → {Range[8], Range[20]}, ImageSize → 500, PlotRange → All]
```



The total time one can expect to be in the system for queue 2 is about .5555556 hours, or about 33 minutes. Suppose you wanted to get that time down to 15 minutes? There are two ways to do that - either get the representatives to serve people more quickly or add more representatives.

To find how many representatives you need to get the time to 15 minutes (or $\frac{1}{4}$ hour), we just set up the queue with *k* channels instead of two:

```
queue2a = QueueingProcess[8, 5, k];
```

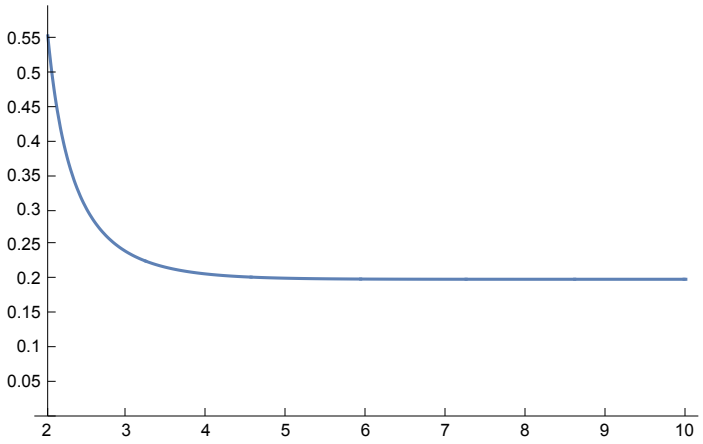
The formula for the mean system time for this queue (as a function of *k*) is:

```
systemtime2a = QueueProperties[queue2a, "MeanSystemTime"]
```

$$\frac{1}{5} + \left(5 \times 8^{k}\, k\, \text{Gamma}[k]\right) \Big/ \left((-8+5k)\left(5 \times 8^{k}\, k\, \text{Gamma}[k] + 5^{k}\, e^{8/5}\, (-8+5k)\, k!\, \text{Gamma}\left[k, \frac{8}{5}\right]\right)\right)$$

Graphing this from $k = 2$ to k = 10 looks like this:

```
Plot[ systemtime2a, {k, 2, 10},
 Ticks → {Range[2, 10], Range[0, 1, .05]}, PlotRange → {0, .6}]
```



From the graph it's clear that $n = 3$ barely works:

```
QueueProperties[ QueueingProcess[8, 5, 3], "MeanSystemTime"]
```

$$\frac{313}{1309}$$

```
N[%]
```

```
0.239114
```

We can also get down to a 15 minute time by getting the 2 representatives to help people more quickly (say n per hour):

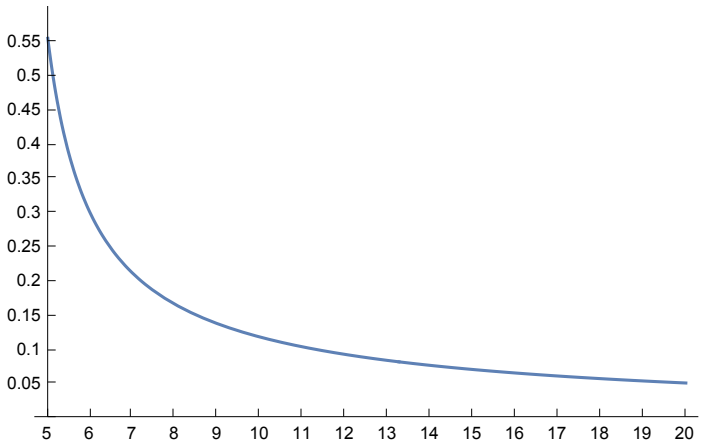
```
queue2b = QueueingProcess[ 8, n, 2];
```

The mean system time in this case is:

```
systemtime2b = QueueProperties[queue2b, "MeanSystemTime"]
```

$$\frac{1}{n} + 16 \Big/ \left((-4+n) \left(32 + e^{8/n} (-4+n)\, n\, \text{Gamma}\left[2, \frac{8}{n}\right] \right) \right)$$

```
Plot[systemtime2b, {n, 5, 20}, Ticks → {Range[5, 20], Range[0, 1, .05]},
 PlotRange → {0, .6}, AxesOrigin → {5, 0}]
```



From this it looks like a service rate of between 6 and 7 people will do the trick, but we can be more precise:

```
FindRoot[systemtime2b == 1 / 4, {n, 7}]
```

```
{n → 6.47214}
```

So two representatives helping about 6.5 people an hour on average will get the average system time down to 15 minutes.

Queue Networks

Mathematica can also stack queues into a network. This works by representing each queue in the network as a "node", and then descrbing the nodes by 4 quantities (or 5 in the case of a "closed network" where there are no external inputs):

A list of arrival rates from outside the system (one for each node, all 0's for a "closed network").
A matrix which represents the probability people will pass between any pair of the nodes.
A list of the service rates for the nodes.
A list of the number of channels (representatives in the previous examples) for each node.
(The number of jobs in the system for a "closed network")

For example suppose that on opening day students arrive on campus at an average rate of 100 per hour. They wander around campus for about a half hour. Then they get their permits from a table staffed by 3 people, each of whom can average 35 people/hour. Then they get their schedule from a table staffed by 5 people, each of whom can handle an average of 25 people/hour. On average how long will it take a student to get through the system?

The nodes in this case are {campus arrivals, permit table, schedule table}. The arrivals (viewed as coming from outside the system) would be {100,0,0}. From "campus arrivals", there is a 100% chance of going to the permit table and a 0% chance of going elsewhere. Likewise from the permit table there is a 100% of going to the schedule table and a 0% chance of going back to campus arrivals or the permit table. And once you are done at the schedule table, you leave the system (not going back to any of the three nodes). Expressed as probabilities instead of percentages this looks like:

```
TableForm[Map[Text, {{0, 1, 0}, {0, 0, 1}, {0, 0, 0}}, {2}],
 TableHeadings → {Text /@ { "Campus arrival", "Permit table", "Schedule table"},
   Text /@ { "Campus arrival", "Permit table", "Schedule table"}}]
```

	Campus arrival	Permit table	Schedule table
Campus arrival	0	1	0
Permit table	0	0	1
Schedule table	0	0	0

In terms of the the average service rates for each channel we have 2/hour for wandering around campus (as the average time is $\frac{1}{2}$hour), 35/hour for the permit table, and 25/hour for the schedule table - for a service list of {2, 35, 25}. In terms of how many channels there are for each node there is no limit for people wandering around campus (so the number of service reps can be said to be ∞), there are three representatives at the permit table, and 5 representatives at the schedule table (for a channel list of {∞, 3, 5}). Defining these in *Mathematica* and combining them with QueuingNetworkProcess now looks like:

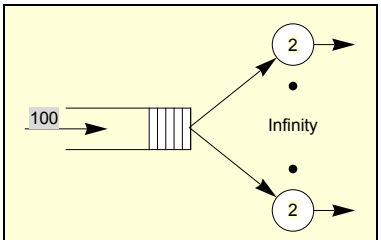
```
arrivals = {100., 0, 0};
passmatrix = {{0, 1, 0}, {0, 0, 1}, {0, 0, 0}};
service = {2, 35, 25};
channels = {∞, 3, 5};

campus = QueueingNetworkProcess[arrivals, passmatrix, service, channels];
```

We can quickly get a look at the properties for each of the three nodes:

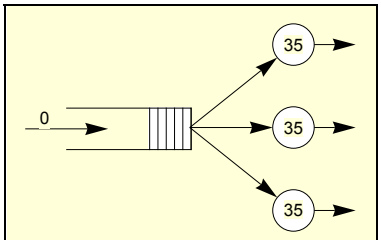
```
Grid[{Text /@ {"Campus arrival", "Permit table", "Schedule table"},
  {QueueProperties[{campus, 1}], QueueProperties[{campus, 2}],
   QueueProperties[{campus, 3}]}}, Alignment → Top]
```

Campus arrival



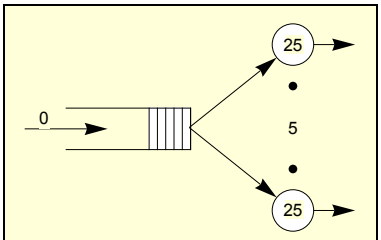
Basic Properties	
NetworkType	Open
NodeCount	3
SelectedNode	1
Performance Measures	
MeanSystemSize	50.
MeanSystemTime	0.5
MeanQueueSize	0.
MeanQueueTime	0.

Permit table



Basic Properties	
NetworkType	Open
NodeCount	3
SelectedNode	2
Performance Measures	
MeanSystemSize	21.0845
MeanSystemTime	0.210845
MeanQueueSize	18.2274
MeanQueueTime	0.182274

Schedule table



Basic Properties	
NetworkType	Open
NodeCount	3
SelectedNode	3
Performance Measures	
MeanSystemSize	6.21645
MeanSystemTime	0.0621645
MeanQueueSize	2.21645
MeanQueueTime	0.0221645

Based on this information we can answer a wide variety of questions:

On average, how many minutes does it take for someone to get through the whole process?

```
totaltime = Sum[QueueProperties[{campus, t}, "MeanSystemTime"], {t, 1, 3}] * 60
```

```
46.3806
```

On average, how long will people be waiting in line?

```
totalwaittime = Sum[QueueProperties[{campus, t}, "MeanQueueTime"], {t, 1, 3}] * 60
```

```
12.2663
```

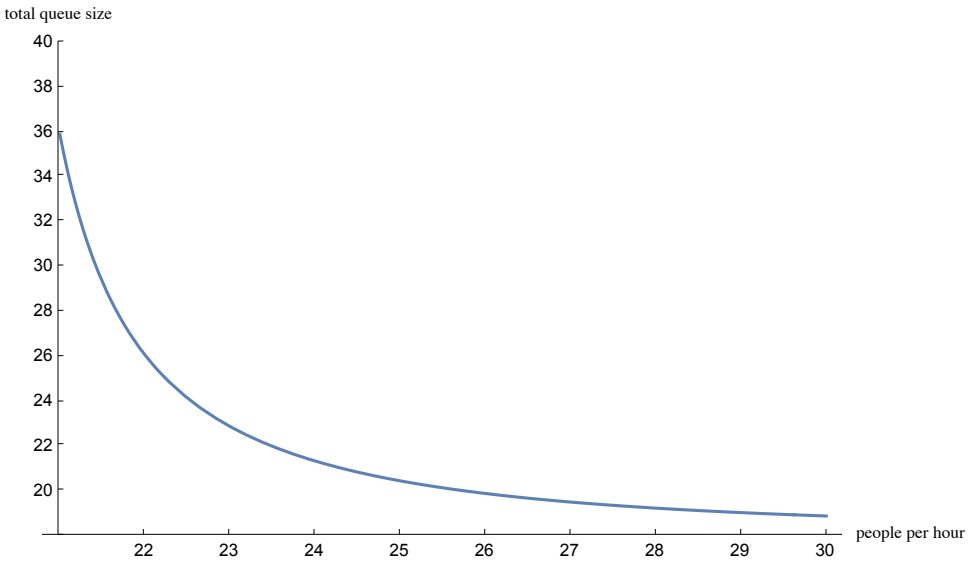
On average how many people will there be wandering around on campus, being helped, or waiting in line?

```
totalpeople = Sum[ QueueProperties[{campus, t}, "MeanSystemSize"], {t, 1, 3}]
```

```
77.301
```

How would the number of people waiting in line change if you can change the speed with which people at the scheduling table can help people?

```
Plot[ Evaluate[ Sum[ QueueProperties[
    {QueueingNetworkProcess[arrivals, passmatrix, {2, 35, k}, channels], t},
    "MeanQueueSize"], {t, 2, 3}]], {k, 21, 30},
 Ticks → { Range[22, 30], Range[0, 100, 2]}, AxesOrigin → {21, 18},
 PlotRange → {18, 40},
 AxesLabel → Text /@ {"people per hour", "total queue size"} , ImageSize → 500]
```



Note that around the original service rate of 25 people per hour the graph is fairly flat so there is not much improvement to be had by making the exsting people more efficient - most of the possible gains are already had by getting up to 25 people per hour.

Author: Dr. Christopher Moretti
Revision Date: 8/6/2015