Simulating a doctor's waiting room

Goal: Simulate the queuing behavior at a doctor's office.

Assumptions:

1. Patients arrive on the minutes between 9:00 am and 11:59 am.

2. At most one patient arrives during any minute.

3. The probability that a patient arrives in any one minute is 0.075.

4. Each patient needs to meet with the doctor for 15 minutes.

Expectations:

1. How many patients does the doctor expect to see in the day?

2. How much time does the doctor expect to spend with patients?

Simulating a doctor's waiting room

How do we simulate the arrival of patients?

Assumptions:

1. Patients arrive on the minutes between 9:00 am and 11:59 am.

2. At most one patient arrives during any minute.

3. The probability that a patient arrives in any one minute is 0.075.

We let i be the counter for time. $i = 0$ occurs at time 9:00 am.

$i =$ ___ occurs at time 11:59 am.

We set up a For loop:

```
For[i = 0, i _____, i _____,
   newPatient = If[RandomReal[] <= 0.075, 1, 0] ]
```

Dealing with the waiting

How do we simulate "waiting" in a "waiting room"?

An arriving patient will wait when _____.

We will keep track of the following variables:

$\mathtt{nwait} =$ The number of patients waiting.

$\mathtt{endTime} = \dfrac{\text{The time when the current patient}}{\text{finishes with the doctor.}}$

$\mathtt{busy} = \begin{cases} 1 \text{ if the doctor is busy} \\ 0 \text{ if the doctor is free} \end{cases}$

Simulating a doctor's waiting room

Now translate the flowchart into an algorithm:

Pseudocode:

▶ Zero out the counters.

▶ For `i` from 0 up to 180,

> ▶ If the doctor is finishing with a patient at time `i`, then set `busy=0`.
> ▶ Determine if a new patient arrives (random, probability 0.075).
>   If so, set `newPatient` to 1; otherwise set `newPatient` to 0.
> ▶ If `newPatient == 1`, add one to the number waiting (`nwait++`)
> ▶ If the doctor is not busy AND there is a patient waiting,
>
> > ▶ Subtract one from the number waiting (`nwait--`)
> > ▶ Set the doctor to be busy (`busy = 1`)
> > ▶ Set the time when the doctor is not busy (`endTime = i + 15`).

Dealing with the waiting

```
nwait = 0; busy = 0; endTime = 0;
For[i = 0, i < 180, i++,
   If[endTime == i, busy = 0];
   newPatient = If[RandomReal[] <= 0.075, 1, 0];
   If[newPatient == 1, nwait++];
   If[busy == 0 && nwait > 0,
      nwait--; busy = 1; endTime = i + 15];
]
```

▶ For i from 0 up to 180,

  ▶ If the doctor is finishing with a patient at time i, then set busy=0.
  ▶ Determine if a new patient arrives (random, probability 0.075).
    If so, set newPatient to 1; otherwise set newPatient to 0.
  ▶ If newPatient == 1, add one to the number waiting (nwait++)
  ▶ If the doctor is not busy AND there is a patient waiting,

    ▶ Subtract one from the number waiting (nwait--)
    ▶ Set the doctor to be busy (busy = 1)
    ▶ Set the time when the doctor is not busy to $i + 15$.

What does the simulation tell us?

We did the simulation, but what was the point?

- ▶ How much of the day will the doctor will be busy?
- ▶ What is the average number of patients in the waiting room?
- ▶ How many people are in the waiting room at noon?
- ▶ How late will the doctor stay after noon?

What statistics do we need to keep track of to answer these questions?
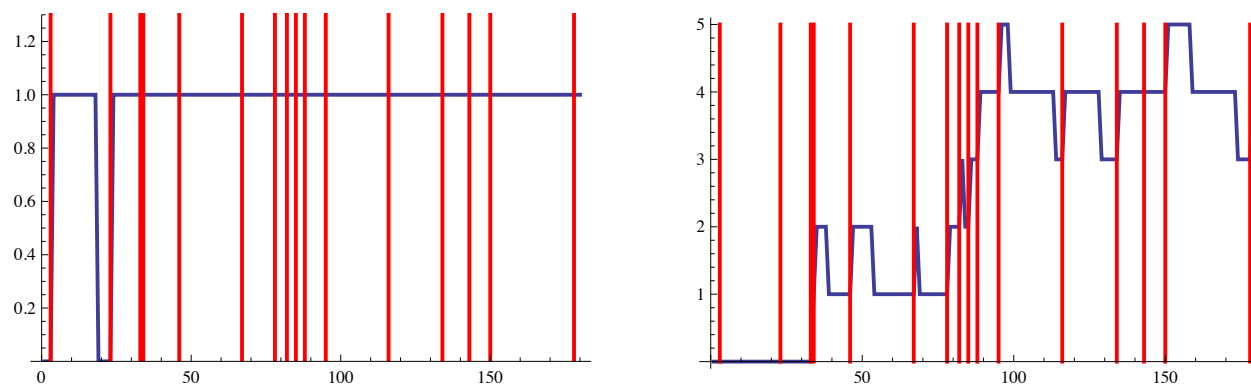
This is just one instance; the power of simulation comes from running the model many times and understanding the average behavior.

Gathering data

▶ How many people are in the waiting room at noon?

```
nwait = 0; busy = 0; endTime = 0;
For[i = 0, i < 180, i++,
   If[endTime == i, busy = 0];
   newPatient = If[RandomReal[] <= 0.075, 1, 0];
   If[newPatient == 1, nwait++];
   If[busy == 0 && nwait > 0,
      nwait--; busy = 1; endTime = i + 15];
   ];
nwait ⟵ [Outputs the value after the loop ends.]
```
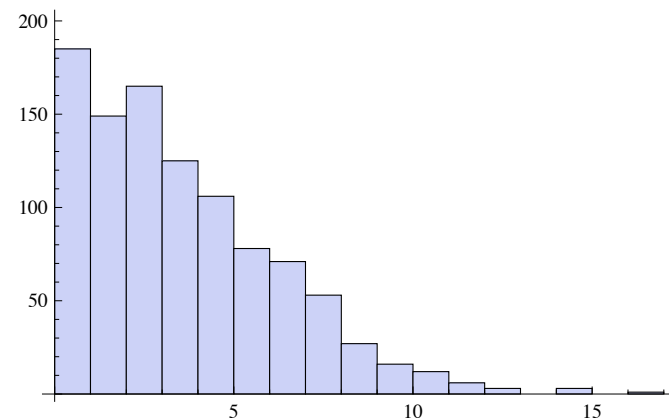
Running many trials

▶ How many people are in the waiting room at noon?

Simulate 1000 times using a `Table` command, generate a histogram.

```
trials = Table[
nwait = 0; busy = 0; endTime = 0;
For[i = 0, i < 180, i++,
   If[endTime == i, busy = 0];
   newPatient = If[RandomReal[] <= 0.075, 1, 0];
   If[newPatient == 1, nwait++];
   If[busy == 0 && nwait > 0,
      nwait--; busy = 1; endTime = i + 15];
   ];
nwait  ], {j,1000}]
```
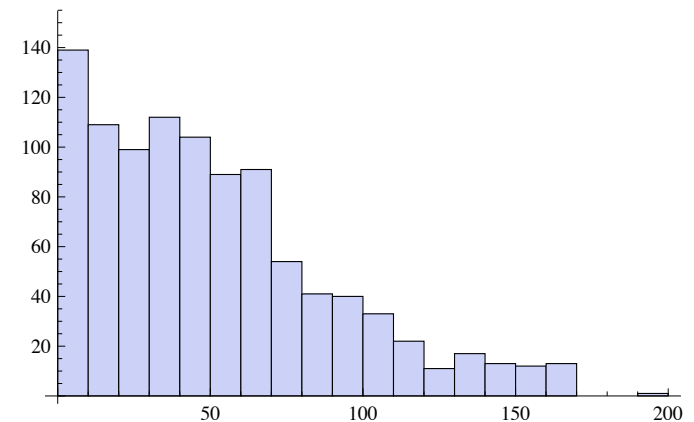
Mean[trials] ⟵ [Average: 3.105]

Histogram[trials]

Gathering data

▶ How late will the doctor stay after noon?

```
nwait = 0; busy = 0; endTime = 0;
For[i = 0, i < 180, i++,
   If[endTime == i, busy = 0];
   newPatient = If[RandomReal[] <= 0.075, 1, 0];
   If[newPatient == 1, nwait++];
   If[busy == 0 && nwait > 0,
      nwait--; busy = 1; endTime = i + 15];
   ];
```
*****

↑ [*How to calculate how long the doctor has to stay after noon?*]

Mean: 51 minutes (does this make sense?)
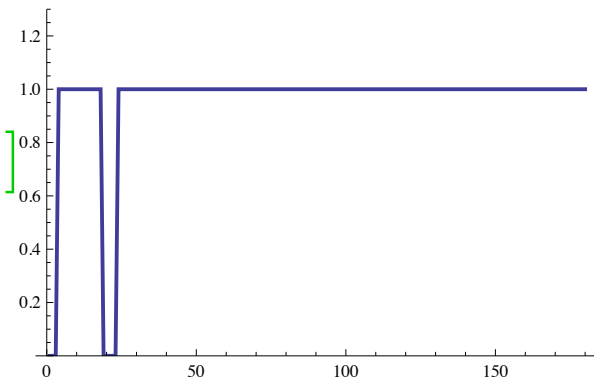
Gathering and plotting time-dependent data

▶ How much of the day will the doctor will be busy?

Keep track of if the doctor is busy by using a variable `isBusy`.

```
nwait = 0; busy = 0; endTime = 0;
For[i = 0, i < 180, i++,
   If[endTime == i, busy = 0];
   newPatient = If[RandomReal[] <= 0.075, 1, 0];
   If[newPatient == 1, nwait++];
   If[busy == 0 && nwait > 0,
      nwait--; busy = 1; endTime = i + 15];
   isBusy[i]=busy;  ⟵ [this copies busy into isBusy[i].]
];         ↓ [Puts data into a list.]
busyList=Table[isBusy[i],i,0,179]
Total[busyList]  ⟵ [Total time busy.]
ListLinePlot[busyList]
```

Gathering and plotting time-dependent data

▶ What is the average number of patients in the waiting room?

Keep track of number of waiting patients by using a variable `numWait`.

```
nwait = 0; busy = 0; endTime = 0;
For[i = 0, i < 180, i++,
   If[endTime == i, busy = 0];
   newPatient = If[RandomReal[] <= 0.075, 1, 0];
   If[newPatient == 1, nwait++];
   If[busy == 0 && nwait > 0,
      nwait--; busy = 1; endTime = i + 15];
   numWait[i]=nwait;  ⟵ [this copies nwait into numWait[i].]
];              ↓ [Puts data into a list.]
waitList=Table[numWait[i],i,0,179]
m=Mean[waitList] ⟵ [Average patients.]
p1=ListLinePlot[waitList];
p2=Plot[m,x,0,179];
Show[p1,p2]
```