

# Problem 79

A common security method used for online banking is to ask the user for three random characters from a passcode. For example, if the passcode was 531278, they may ask for the 2nd, 3rd, and 5th characters; the expected reply would be: 317.

The text file, `keylog.txt`, contains fifty successful login attempts.

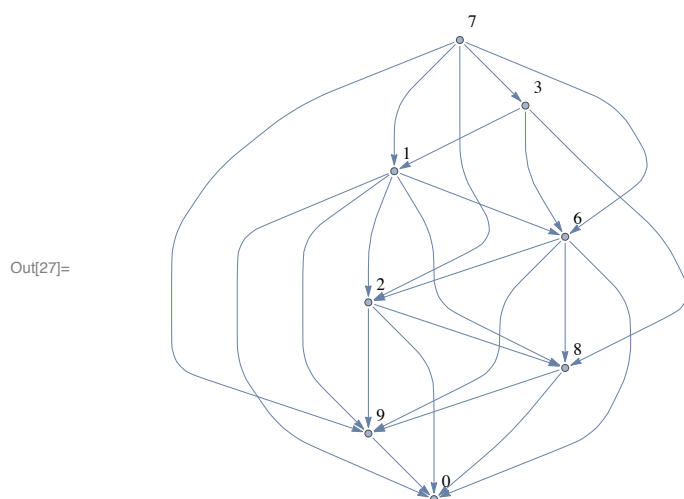
Given that the three characters are always asked for in order, analyse the file so as to determine the shortest possible secret passcode of unknown length.

## Solution

```
In[24]:= data = {319, 680, 180, 690, 129, 620, 762, 689, 762, 318, 368,  
              710, 720, 710, 629, 168, 160, 689, 716, 731, 736, 729, 316, 729,  
              729, 710, 769, 290, 719, 680, 318, 389, 162, 289, 162, 718, 729,  
              319, 790, 680, 890, 362, 319, 760, 316, 729, 380, 319, 728, 716};
```

We'll construct a graph where  $a \rightarrow b$  if  $b$  follows  $a$  somewhere in the code, and then find a Hamiltonian path.

```
In[27]:= Graph[DeleteDuplicates@
  Flatten[(DirectedEdge @@@ Partition[IntegerDigits[#, 2, 1]) & /@ data],
  VertexLabels -> "Name", ImagePadding -> 40]
```



```
In[28]:= Graph[Join[{0 → 7}, DeleteDuplicates@
    Flatten[(DirectedEdge @@@ Partition[IntegerDigits[#, 2, 1]) & /@ data]],
    VertexLabels → "Name", ImagePadding → 40] // FindHamiltonianCycle
```

```
Out[28]= { { 0 → 7, 7 → 3, 3 → 1, 1 → 6, 6 → 2, 2 → 8, 8 → 9, 9 → 0 } }
```

And we started at 7, so the answer is just:



```
In[29]:= Last /@First@%
```

```
Out[29]= { 7, 3, 1, 6, 2, 8, 9, 0 }
```