

### 3. You Can Read Minds (With a Little Calibration)

*Tact is after all a kind of mind reading. – Sarah Orne Jewett.*

Programming constructs and algorithmic paradigms covered in this puzzle:  
Reading input from a user, and control flow for case analysis. Encoding and decoding information.

You are a Magician and a Mind Reader Extraordinaire. The Assistant goes into the audience with an authentic deck of 52 cards while you are outside the room and can't possibly see anything. Five audience members each select one card from the deck. The Assistant then gathers up the five cards. The Assistant shows the entire audience four cards, one at a time. For each of these four cards, the Assistant asks the audience to mentally focus on the card, while you look away and try to read their collective minds. Then, after a few seconds you are shown the card. This helps you calibrate your mind reading to the particular audience ☺

After you see these four cards, you claim that you are well calibrated to this audience and leave the room. The Assistant shows the fifth card to the audience and puts it away. Again, the audience mentally focuses on the fifth card. You return to the room, concentrate for a short time and correctly name the secret, fifth card!

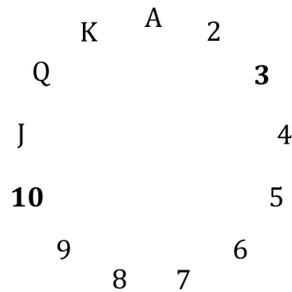
You are in cahoots with your Assistant and have planned and practiced this trick. However, everyone is watching closely and the only information that the Assistant can give you is the four cards.

*How does this trick work?*

It turns out that the order in which the Assistant reveals the cards tells the Magician what the fifth card is! The Assistant needs to be able to decide which of the cards is going to be hidden – he or she cannot allow the audience to pick the hidden card out of the five cards that the audience picks. Here’s one way that the Assistant and the Magician can work together.

As a running example, suppose that the audience selects:  $10\heartsuit 9\spadesuit 3\heartsuit Q\clubsuit J\diamonds$

- The Assistant picks out two cards of the same suit. Given five cards, there will *definitely* be two with the same suit, since we only have four different suits. In the example, the assistant might choose the  $3\heartsuit$  and  $10\heartsuit$ . (Mathematically inclined readers may recognize this as the Pigeonhole Principle. Given  $n$  holes and  $n + 1$  pigeons that each need to fly through some hole, there is at least one hole through which two pigeons will fly.)
- The Assistant locates the values of these two cards on the cycle of cards shown below:



For any two distinct values on this cycle, one is always between 1 and 6 hops clockwise from the other. For example, though the  $10\heartsuit$  is 7 hops clockwise from the  $3\heartsuit$ , the  $3\heartsuit$  is 6 hops clockwise from the  $10\heartsuit$ .

- One of these two cards is revealed first, and the other becomes the secret card. The card that is revealed is the card *from* which we can reach the other card clockwise in 6 or fewer hops. Thus, in our example, the  $10\heartsuit$  would be revealed, and the  $3\heartsuit$  would be the secret card, since we can reach the  $3\heartsuit$  from the  $10\heartsuit$  in 6 hops. (If on the other hand the two cards were the  $4\heartsuit$  and the  $10\heartsuit$ , the  $4\heartsuit$  would be revealed since the  $10\heartsuit$  is 6 hops clockwise from the  $4\heartsuit$ .)
  - The suit of the secret card is the same as the suit of the first card revealed.
  - The value of the secret card is between 1 and 6 hops clockwise from the value of the first card revealed.
- All that remains is to communicate a number between 1 and 6. The Magician and Assistant agree beforehand on an ordering of all the cards in the deck from smallest to largest such as:

$A\clubsuit A\diamonds A\heartsuit A\spadesuit 2\clubsuit 2\diamonds 2\heartsuit 2\spadesuit \dots Q\clubsuit Q\diamonds Q\heartsuit Q\spadesuit K\clubsuit K\diamonds K\heartsuit K\spadesuit$

The order in which the last three cards are revealed communicates the number according to the following scheme:

( small, medium, large ) = 1  
( small, large, medium ) = 2  
( medium, small, large ) = 3  
( medium, large, small ) = 4  
( large, small, medium ) = 5  
( large, medium, small ) = 6

In the example, the Assistant wants to send 6 and so reveals the remaining three cards in large, medium, small order. Here is the complete sequence that the Magician sees: 10♥ Q♠ J♦ 9♦

- The Magician starts with the first card, 10♥, and hops 6 values clockwise to reach 3♥, which is the secret card!

## Exercises

**Exercise 1:** There is a small bug in the ComputerAssistant program because we got lazy with the following code:

```
7.         for i in range(5):
8.             number = number * (i + 1) // (i + 2)
9.             n = number % 52
```

We are using an input number to generate five “random” cards. The problem with this particular strategy is that it does not check that the five cards obtained are distinct. In fact, if you input the number 888888, the following five cards are generated in this order: ['A\_C', 'A\_C', '7\_H', 'J\_D', 'K\_S']. You can see the problem here – we are not checking for duplicate cards in ComputerAssistant. Fix this procedure to check for distinct cards, and generate additional “random” numbers by running the loop above for additional iterations till five distinct cards are generated.

**Exercise 2:** Modify ComputerAssistant so in the case of two pairs of cards with the same suit, the hidden and first cards are selected such that the number to be encoded is the smallest possible.

**Exercise 3:** Some magicians prefer a different ordering of the cards where the suit of the card is the major determinant of the order as opposed to the number as shown below.

```
deck = ['A_C', '2_C', '3_C', '4_C', '5_C', '6_C', '7_C', '8_C',
        '9_C', '10_C', 'J_C', 'Q_C', 'K_C', 'A_D', '2_D', '3_D',
        '4_D', '5_D', '6_D', '7_D', '8_D', '9_D', '10_D', 'J_D',
        'Q_D', 'K_D', 'A_H', '2_H', '3_H', '4_H', '5_H', '6_H',
```

```
'7_H', '8_H', '9_H', '10_H', 'J_H', 'Q_H', 'K_H', 'A_S',  
'2_S', '3_S', '4_S', '5_S', '6_S', '7_S', '8_S', '9_S',  
'10_S', 'J_S', 'Q_S', 'K_S']
```

Modify ComputerAssistant so the Magician can practice with the above order. Note that the computation of numbers and suits of the cards given the index of the card needs to change. And, of course, the ordering of cards read out by the Assistant will change as a result. A few details to get right in this variant solution to the puzzle!

MIT OpenCourseWare  
<https://ocw.mit.edu>

6.S095 Programming for the Puzzled  
January IAP 2018

For information about citing these materials or our Terms of Use, visit: <https://ocw.mit.edu/terms>.