# Solitaire Cipher Class Exercise

*CS461 Fall 2009*
*September 14*
*With edits*

Today you will use a deck of cards to encipher and decipher messages using the Solitaire Cipher created by Bruce Schneier http://www.schneier.com/solitaire.html. This is a stream cipher that uses a deck of cards to create a key stream. It uses an output feedback mode to create the elements of the key stream. The card deck acts as the register, and the solitaire operations act as the encryption method that mixes the register.

## Class Exercise Goals

The goal of this exercise is to become familiar with the solitaire key generation algorithm. Schneier's article (http://www.schneier.com/solitaire.html) gives the details. The highlights of the steps are outlined in this document.

Schneier's article shows the state of the deck (register) between each step for the first two key generations (repeated at the end of this document). Start with his "unshuffled" deck ordering (A-Kc, A-Kd, A-Kh, A-Ks, Ja, Jb) and work through a couple keys to make sure you have the steps correct.

Then re-key your deck as (A-Ks, A-Kc, Ja, A-Kh, Jb, A-Kd). Generate sufficient keys to decrypt: BTURAI. Note, this was encrypted using the tableau at the end of this document. It disagrees with the encryption/decryption algorithm that Scheier uses. The tableau assumes the keys start with 0. The algorithms rightly assume the keys start with 1. Decrypt using the tableau and you should retrieve the value I started with.

## Card Values

The cipher requires a full deck plus two jokers. The jokers must be distinguishable and are referenced in the algorithm as Joker A and Joker B.

Map the cards to numbers as follows:

- Clubs 1 through King = 1- 13
- Diamonds 1 through King = 14 – 26
- Hearts 1 through King = 27 – 39
- Spades 1 through King = 40 – 52.

## Initializing the deck

The initial order of the card deck must be identical between the encrypter and the decrypter. Schneier's article discusses several ways of doing this. Ideally, you can do a good shuffle as the encrypter and then recreate the original shuffled order for you partner. The article also discusses using a pass phrase to initialize the deck. We'll use a couple different orders in our exercise as discussed below. We'll be working with the deck face up.

## Creating the Key Stream

For each key element perform the following steps:

1. Find Joker A, move it back one card.
2. Find Joker B, move it back two cards.
3. Take all cards in front of the “first” Joker (either A or B), and switch them with all the cards after the “second” Joker (either A or B).
4. Look at the last card. Convert to the numeric value and count that many cards back from the top of the deck. Either Joker is valued at 53. Take those cards and insert at the back of the deck in front of the last card.
5. Look at the top card. Convert to the numeric value (again with either Joker valued at 53) and count that many cards back from the top of the key. This is the key value
   - If the key card is Joker, no key value is generated in this round.

## Encrypt and Decrypt

Once the key stream is generated, combine the plaintext with the key stream using the standard character shifting operation used for the Caesar, Vinegere, and book ciphers.

E(pi) = (pi + ki) mod 26 = ci

D(ci) = (52 + ci – ki) mod 26 = pi

Alternatively, you can use the variant of the Vigenere tableau on the next page. The key values are listed on the left hand side, the plaintext characters across the top, and the ciphertext values in the table.

### Schneier's Unshuffled example

Sample 1: Start with an unkeyed deck: A-clubs to K-clubs, A-diamonds to K-diamonds, A-hearts to K-hearts, A-spades to K-spades, A joker, B joker. (You can think of this as 1 ... 52, A, B.)

Here's how to generate the first two outputs. The initial deck is:

```
1 2 3 4 ... 52 A B
```

After the first step (moving the A joker):

```
1 2 3 4 ... 52 B A
```

After the second step (moving the B joker):

```
1 B 2 3 4 ... 52 A
```

After the third step (the triple cut):

```
B 2 3 4 ... 52 A 1
```

After the fourth step (the count cut):

```
2 3 4 ... 52 A B 1
```

The last card is a 1, which means cut one card. Remember that the 1 stays where it is, so the one card (the B, moves to the bottom of the deck just above the 1.

The fifth step does not change the deck, but produces an output card. The top card is a 2, so count down two cards to the 4. The first Solitaire output is 4. (Of course, you're not supposed to remove this card from the deck. Keep the 4 where it is; just write it down somewhere.)

To produce the second Solitaire output, go through the five steps again.

Step 1:

```
2 3 4 ... 49 50 51 52 B A 1
```

Step 2:

```
2 3 4 ... 49 50 51 52 A 1 B
```

Step 3:

```
A 1 B 2 3 4 ... 49 50 51 52
```

Step 4:

```
51 A 1 B 2 3 4 ... 49 50 52
```

The last card is a 52, so count 52 cards down to the 51. Cut the single card, the 51 with the rest of the deck. Remember that the 52 remains unmoved.

Step 5 produces the output card. The first card is a 51. Counting down fifty-one cards gets to the 49, which is the second output card. (Again, don't remove the 49 from the deck.)

The first ten outputs are:

```
4 49 10 (53) 24 8 51 44 6 4 33
```

The 53 is skipped, of course. I just put it there for demonstration.

If the plaintext is

```
AAAAA  AAAAA
```

then the ciphertext is:

```
EXKYI  ZSGEH
```

| | a | b | c | d | e | f | g | h | i | j | k | l | m | n | o | p | q | r | s | t | u | v | w | x | y | z |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1c/1h | a | b | c | d | e | f | g | h | i | j | k | l | m | n | o | p | q | r | s | t | u | v | w | x | y | z |
| 2c/2h | b | c | d | e | f | g | h | i | j | k | l | m | n | o | p | q | r | s | t | u | v | w | x | y | z | a |
| 3c/3h | c | d | e | f | g | h | i | j | k | l | m | n | o | p | q | r | s | t | u | v | w | x | y | z | a | b |
| 4c/4h | d | e | f | g | h | i | j | k | l | m | n | o | p | q | r | s | t | u | v | w | x | y | z | a | b | c |
| 5c/5h | e | f | g | h | i | j | k | l | m | n | o | p | q | r | s | t | u | v | w | x | y | z | a | b | c | d |
| 6c/6h | f | g | h | i | j | k | l | m | n | o | p | q | r | s | t | u | v | w | x | y | z | a | b | c | d | e |
| 7c/7h | g | h | i | j | k | l | m | n | o | p | q | r | s | t | u | v | w | x | y | z | a | b | c | d | e | f |
| 8c/8h | h | i | j | k | l | m | n | o | p | q | r | s | t | u | v | w | x | y | z | a | b | c | d | e | f | g |
| 9c/9h | i | j | k | l | m | n | o | p | q | r | s | t | u | v | w | x | y | z | a | b | c | d | e | f | g | h |
| 10c/10h | j | k | l | m | n | o | p | q | r | s | t | u | v | w | x | y | z | a | b | c | d | e | f | g | h | i |
| Jc/Jh | k | l | m | n | o | p | q | r | s | t | u | v | w | x | y | z | a | b | c | d | e | f | g | h | i | j |
| Qc/Qh | l | m | n | o | p | q | r | s | t | u | v | w | x | y | z | a | b | c | d | e | f | g | h | i | j | k |
| Kc/Kh | m | n | o | p | q | r | s | t | u | v | w | x | y | z | a | b | c | d | e | f | g | h | i | j | k | l |
| 1d/1s | n | o | p | q | r | s | t | u | v | w | x | y | z | a | b | c | d | e | f | g | h | i | j | k | l | m |
| 2d/2s | o | p | q | r | s | t | u | v | w | x | y | z | a | b | c | d | e | f | g | h | i | j | k | l | m | n |
| 3d/3s | p | q | r | s | t | u | v | w | x | y | z | a | b | c | d | e | f | g | h | i | j | k | l | m | n | o |
| 4d/4s | q | r | s | t | u | v | w | x | y | z | a | b | c | d | e | f | g | h | i | j | k | l | m | n | o | p |
| 5d/5s | r | s | t | u | v | w | x | y | z | a | b | c | d | e | f | g | h | i | j | k | l | m | n | o | p | q |
| 6d/6s | s | t | u | v | w | x | y | z | a | b | c | d | e | f | g | h | i | j | k | l | m | n | o | p | q | r |
| 7d/7s | t | u | v | w | x | y | z | a | b | c | d | e | f | g | h | i | j | k | l | m | n | o | p | q | r | s |
| 8d/8s | u | v | w | x | y | z | a | b | c | d | e | f | g | h | i | j | k | l | m | n | o | p | q | r | s | t |
| 9d/9s | v | w | x | y | z | a | b | c | d | e | f | g | h | i | j | k | l | m | n | o | p | q | r | s | t | u |
| 10d/10s | w | x | y | z | a | b | c | d | e | f | g | h | i | j | k | l | m | n | o | p | q | r | s | t | u | v |
| Jd/Js | x | y | z | a | b | c | d | e | f | g | h | i | j | k | l | m | n | o | p | q | r | s | t | u | v | w |
| Qd/Qs | y | z | a | b | c | d | e | f | g | h | i | j | k | l | m | n | o | p | q | r | s | t | u | v | w | x |
| Kd/Ks | z | a | b | c | d | e | f | g | h | i | j | k | l | m | n | o | p | q | r | s | t | u | v | w | x | y |