The Challenge: Secret Message - Cryptography

This problem is inspired by none other than everyone's favorite spy ~~Bon, James Bon~~. Sorry Bond, James Bond 007 of ~~her~~ His Majesty's secret service.

As a secret agent, you need a method to transmit a message to another secret agent. But an encrypted text written on a notebook will be suspicious if you get caught. A simple deck of playing cards, *and we all know how much ~~Bon~~ Bond loves the casino*, is everything but suspicious...

With a deck of 52 playing cards, there are 52! different possible permutations to order it. And 52! is equal to 80658175170943878571660636856403766975289505440883277824000000000000. That's a number with 68 digits!

There are so many different possible permutations, we can affirm that if you shuffle the cards well and put them back together to form a deck, you are the first one in history to get this particular order. The number of possible permutations in a deck of cards is higher than the estimated number of atoms in planet Earth (which is a number with about 50 digits).

With a way to associate a permutation of the cards to a sequence of characters, we can hide a message in the deck by ordering it correctly.

Correspondence between message and permutation

Message

To compose our message, we will use an alphabet containing 27 characters: the space and the letters from A to Z. We give them the following values:

" " = 0, A = 1, B = 2, ..., Z = 26

We now have a numeral system with a base equal to 27. We can compute a numeric value corresponding to any message:

"A " = 27 "AA" = 28 "AB" = 29 "ABC" = 786 etc.

Permutation

Now we need a way to attribute a unique number to each of the possible permutations of our deck of playing cards.

There are few methods to enumerate permutations and assign a number to each of them, we will use the lexicographical order. With three cards, A♣, 2♣, and 3♣, as an example, it gives:

"A♣","2♣","3♣" = 0

"A♣","3♣","2♣" = 1

"2♣","A♣","3♣" = 2

"2♣","3♣","A♣" = 3

"3♣","A♣","2♣" = 4

"3♣","2♣","A♣" = 5

So, the first arrangement is A♣,2♣,3♣, and the last one is 3♣,2♣,A♣.

With our 52 playing cards – ranks sorted from the Ace to the King, and suits in alphabetical order (Clubs, Diamonds, Hearts, Spades) – the first arrangement (number 0) is:

Ace of Clubs to King of Clubs, then Ace of Diamonds to King of Diamonds, then Ace of Hearts to King of Hearts, then Ace of Spades to King of Spades.

and the last one (number 52! - 1) is:

King of Spades to Ace of Spades, then King of Hearts to Ace of Hearts, then King of Diamonds to Ace of Diamonds, then King of Clubs to Ace of Clubs.

To transmit a message, we will compute the permutation for which the unique number is the numeric value of the message.

Your task

Write two functions:

encode = ("message string" as string*) => ["D", "E", "C", "K"]*

Takes a message (as a string) as an argument and return a deck of playing cards (as an array of strings) representing the 52 cards in the deck as shown below, ordered to hide the message.

decode = (["D", "E", "C", "K"] array of strings*) => "message string"*

Takes a deck of playing cards (as an array of strings) as an argument and returns the message (as a string) hidden inside.

Each card name, in a deck, is written with a two-character string: the rank followed by the suit. The arrangement of the deck is represented as:

AC 2C 3C 4C 5C 6C 7C 8C 9C TC JC QC KC for ♣
AD 2D 3D 4D 5D 6D 7D 8D 9D TD JD QD KD for ♦
AH 2H 3H 4H 5H 6H 7H 8H 9H TH JH QH KH for ♥
AS 2S 3S 4S 5S 6S 7S 8S 9S TS JS QS KS for ♠

The alphabet is represented as “ ABCDEFGHIJKLMNOPQRSTUVWXYZ”

Please note the space “ “ at the beginning of the alphabet string.

All letters are uppercase only. There is no punctuation.

The array will always contain 52 elements (A full deck)

Examples

Encoding

encode("Hello World")

Returns:

[

“AC”, “2C”, “3C”, “4C”, “5C”, “6C”, “7C”, “8C”, “9C”, “TC”, “JC”, “QC”, “KC”,
“AD”, “2D”, “3D”, “4D”, “5D”, “6D”, “7D”, “8D”, “9D”, “TD”, “JD”, “QD”, “KD”,
“AH”, “2H”, “3H”, “4H”, “5H”, “6H”, “7H”, “8H”, “KH”, “9S”, “8S”, “3S”, “5S”,
“4S”, “QS”, “TH”, “6S”, “QH', '2S', '9H', 'AS', '7S', 'JH', 'KS', “TS”, “JS”

]

Decoding

Decode([

"AC", "2C", "3C", "4C", "5C", "6C", "7C", "8C", "9C", "TC", "JC", "QC", "KC",

"AD", "2D", "3D", "4D", "5D", "6D", "7D", "8D", "9D", "TD", "JD", "QD", "KD",

"AH", "2H", "3H", "4H", "8H", "9S", "3S", "2S", "8S", "TS", "QS", "9H", "7H",

"KH", "AS", "JH", "4S", "KS", "JS", "5S", "TH", "7S", "6S", "5H", "QH", "6H"

])

Returns:

"ATTACK APPROVED"

Inputs & Outputs

You can assume all test cases are valid.

An encode must return a full deck of cards. Any whitespace starts at element 0.

Input:

A string of alphabetic letters and spaces

OR

A 52 element array

Output:

The encoded 52 element array

OR

The Decoded Array as a String

Scoring

The solution must pass all tests for the score to be considered.

Test program

Test cases [cards.txt file] is provided to help prove out your code.

Any script that passes the test cases can be submitted. If you are surprised that your solution passed the test cases, please submit it anyway!

During the private phase of contest, please email all entries to the judging team directly.

Alexander.Harken@aa.com; Antoine.Westerhof@aa.com; Wayne.Kimball@aa.com

One of us will post your score to the Slack channel once we confirm it passes.

Coding GOLF - How To Play

Most importantly, anybody can play

The object in "real" golf is to hit the ball in the hole in the fewest strokes. The object in Coding Golf is to get from input (tee) to target (hole) in the fewest keystrokes.

Example: How many positive elements are in array $a?

Array $a could be of structure $a=[1, 2 ,-3, 4, -5, -7, 8, 9]

One approach:

```
for $b=0 to ubound($a)
  if $a[$b]>0
    $c=$c+1
  endif
next
```

for a score of 61.

Another solution is:

```
DO
  $b=$b+1
  if $a[$b]>0
    $c=$c+1
  endif
UNTIL $b>(UBOUND($a)+1)
```

for a score of 70.

Better approach: Code sample 1

Coding GOLF - The Rules

1) The goal of Coding Golf is to score the lowest (key)strokes.
2) This challenge must be done in **Python.**
3) Strokes are all characters in a piece of code including space and carriage returns.
    a. Example:
        i. *$arr.vortex = [1,1,1,1] would have a score of 23*
        ii. *$=[1,1,1,1] would have a score of 11*
4) Code can be constructed any way you like if it does not generate syntax or other errors when running the script.
5) The final solution MUST pass all test scripts that are part of the Coding golf challenge.
6) During the private coding phase, no code is allowed to be posted. Violations result in disqualification of said player.
7) During the public coding phase, code should be posted, reused, and borrowed from other players.
8) You can submit scores as often as you want.
9) If you get a score lower than what has been posted, you are obligated to post your score immediately so others can try to compete with you.
10) You may assume ASCII as character set.
11) Tokenizing the script, or portions thereof is not allowed.
12) External Libraries are not allowed, all data manipulation must be present in your script.
    a. For this challenge the Math Library is permitted
13) If something is not explicitly denied by the rules, it's allowed.
14) If Confusion arises, arranger of the Golf round has the final say.

Coding GOLF - The Duration of the Competition

1) The challenge will be released at noon Central on June 9th.
2) You will have till noon June 21st for the private phase.
3) At noon June 21st, the public phase will begin and will continue till noon June 28th.
4) On June 28th at 1:00 Central the winner will be declared.

Most importantly HAVE FUN!