Orasi Monthly Code Challenge

Gold League

June 2013

Challenge Submitted by:

Matt Watson

Challenge Winner:

Lewis Gordon

June 2013 Orasi Code Challenge - Gold

Challenge Description

In cryptography, encryption is the process of encoding messages (or information) in such a way that eavesdroppers or hackers cannot read it, but that authorized parties can. In an encryption scheme, the message or information, referred to as plaintext, is encrypted using an encryption algorithm, turning it into an unreadable ciphertext. For simple encryption the ciphertext can be converted back to plaintext by reverse engineering the encryption algorithm. The Orasi Monthly Code Challenge is to decode a simple encrypted message.

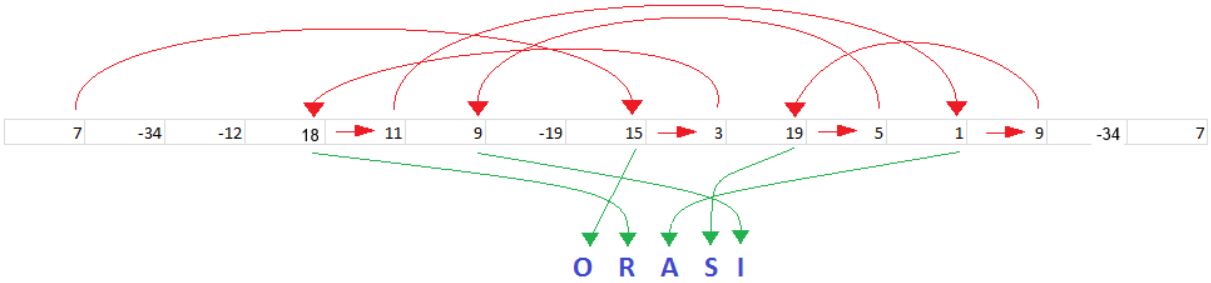
Encryption Algorithm

A secret message has been encrypted and stored in a jagged array. This message has been encrypted using the following rules:

- Each word has been converted in to a series of numbers and stored in a row of the array.
- The sum of the digits in the row will be equal to the length of the word.
- The rows of the array have been put in a random order.
- When the rows are placed in the correct order, the digit in the last column of each row will be in ascending order.
- The number in the first column of the array will be equal to the column index of the first Decryption Number.
- Decryption Numbers correspond with position in the alphabet
    - IE 1 = a, 2 = b, 20 = t.
- The next column after a Decryption number gives the column index of the next Decryption Number.
- The first letter found will be the first letter of the word, the second will be the second letter of the word…
- Values in the array may be positive, negative, or zero.

Examples

Following examples are provided to help make sense of the complex rules.



The following array can be used to test your function

6	-23	5	11	-15	12	13	2	-19	-26	7	19	13	19	15	1	10	-12	-56	5	20
12	-13	27	-9	5	-8	-19	-5	19	-3	-7	-12	1	13							
6	-23	8	4	9	10	20	2	-19	-18	19	-16	2								
4	5	12	-27	19	1	20	-19	18	15	-33	-20	3	8	-26	5	6	15			
5	-23	19	5	-11	9	2	-13	9												

Submission Requirements:

Your submission must be in the form of a VBScript function that can run inside QTP. The function should accept one argument (a jagged array) and return a string of the un-encrypted message. The function name should be your name. This function must comply with Orasi Coding Standards. You will be allowed 5 minutes of execution time to accomplish this task. Anyone who successfully returns the correct message will be considered to have completed the challenge. The code that completes the fastest and meets coding standards will be the winner of the challenge.

Winning Solution:

```
'*************************************************************************************
'Function Name                  DecryptArray
'Author:                        Lewis Gordon
'Created Date:                  6/14/2013
'Purpose:                       Reads an encrypted message in jagged array form and returns
'                               un- encrypted message put string
'*************************************************************************************
Public Function LewisGordon(JaggedArray)
        Dim ArrayToInsert
        Dim Alphabet
        Dim i
        Dim j
        Dim EncryptedRow
        Dim HolePosition
        Dim Message
        Dim NextIndex
        Dim Sum
        Dim Word
```

```
	'Alphabet used as a lookup later on.
	Alphabet = Array("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")


	'Insertion sort to sort the jagged array.
	For i=1 To UBound(JaggedArray)
		HolePosition = i
		ArrayToInsert = JaggedArray(HolePosition)

		Do While HolePosition > 0 And ArrayToInsert(UBound(ArrayToInsert)) < JaggedArray(HolePosition-
1)(UBound(JaggedArray(HolePosition-1)))
			JaggedArray(HolePosition) = JaggedArray(HolePosition-1)
			HolePosition = HolePosition - 1
			If HolePosition = 0 Then
				Exit Do
			End If
		Loop

		JaggedArray(HolePosition) = ArrayToInsert
	Next

	'Decrypt each row and add it to the message.
	For i=0 To UBound(JaggedArray)
		EncryptedRow = JaggedArray(i)
		Word = ""
		Sum = 0
		For j=0 To UBound(EncryptedRow)
			Sum = Sum + EncryptedRow(j)
		Next

		NextIndex = EncryptedRow(0)
		For j=1 To Sum
			Word = Word + Alphabet(EncryptedRow(NextIndex)-1)
			NextIndex = EncryptedRow(NextIndex+1)
		Next
		Message = Message + " " + Word
	Next
	LewisGordon = Trim(Message)
End Function
```