Article
Danny Wijnschenk · Nov 6, 2017
3m read

Advent of Code 2016 Day6: Signals and Noise

This is a series of programming challenges for beginners and experienced Caché programmers.

For an introduction : goto to article https://community.intersystems.com/post/advent-code-2016-day1-no-time-ta...

In today's challenge, you have to repair the communication with Santa : messages are coming in, but jammed. By using repetition code (see also https://en.wikipedia.org/wiki/Repetitioncode), you can find out what's in the message : by sending the same message a number of times, you can calculate which characters are most frequent on each position and find the most likely message.

For example, with input like :

```
eedadn
drvtee
eandsr
raavrd
atevrs
tsrnev
sdttsa
rasrtv
nssdts
ntnada
svetve
tesnvt
vntsnd
vrdear
dvrsen
enarar
```

the most common characters in each column are e, a, s, t, e, r.

The full instructions are on http://adventofcode.com/2016/day/6, the input message is on http://adventofcode.com/2016/day/6/input

The second part of the puzzle is exactly the same, but with the least common letter instead of the most frequent. In my code i just need to replace > by <, not a real challenge !

(The least common characters in the same above input would be a, d, v, e, n, t)

Oh well, here is our solution which does the two challenges at the same time :

```
Class AOC2016.Day6 Extends AOC2016.Utils
{

ClassMethod Run(file As %String = "day6.txt")
{
  #dim message, input as %String
  #dim iInput, iLetter, count, position as %Integer
```

```
  #dim firstMessage, secondMessage as %String
  #dim leastLetter, commonLetter, letter as %Char
  #dim leastCount, commonCount as %Integer

  Try {
    Do ..Input(file, .input)
    //count all input letters in each column
    For iInput=1:1:input {
      Set message=$Translate(input(iInput)," ")
      For iLetter=1:1:$Length(message) {
        Set letter = $Extract(message,iLetter)
        Set count(iLetter, letter)=$Increment(count(iLetter,letter))
      }
    }
    //for each column, get the most common and least common letter and add to the mes
sage
    Set (firstMessage, secondMessage) = ""
    Set position = "" For {
      Set position = $Order(count(position)) If position = "" Quit
      Set (commonLetter, commonCount, leastLetter, leastCount) = ""
      Set letter="" For {
        Set letter=$Order(count(position,letter)) If letter="" Quit
        If (count(position,letter)>commonCount)!(commonCount="") {
          Set commonLetter = letter
          Set commonCount = count(position,letter)
        }
        If (count(position,letter)<leastCount)!(leastCount="") {
          Set leastLetter = letter
          Set leastCount = count(position,letter)
        }
      }
      Set firstMessage = firstMessage_commonLetter
      Set secondMessage = secondMessage_leastLetter
    }
  } Catch {
    Write "Error : ",$ZError,!
  }
  Write "Message 1 : ",firstMessage,!
  Write " 2 : ",secondMessage,!
  }

}
```

As always, you can find Bert's and my solution of all the challenges so far here
: https://bitbucket.org/bertsarens/advent2016 and https://github.com/DannyWijnschenk/AdventOfCode2016

Here is the list of all Advent of Code 2016 articles :

#Caché #Code Snippet #Contest #ObjectScript

Source URL:https://community.intersystems.com/post/advent-code-2016-day6-signals-and-noise

---