ANR

PowerShell

Programming with PowerShell
Complex programs

Preparation

For each task in this task series, create a separate PowerShell script with the extension ps1. Execute the scripts by navigating to the directory where the script is located using a PowerShell (as an administrator if necessary).

Task 1

Implement the number guessing game using PowerShell. Feel free to use the speech synthesizer for audio output[1].

Task 2

Implement Caesar encryption as a PowerShell script. In classical Caesar encryption, each letter in the alphabet is moved forward by three positions in the order, i.e. A becomes D, B becomes E and so on. The last letters X, Y, Z are mapped to A, B, C.

In PowerShell, an ASCII character can be converted to its associated ASCII number (decimal) as follows:

- [byte] [char]'A' (returns 65)

The conversion of the ASCII number to the ASCII character works exactly the other way around:

- [char] [byte] 65 (returns 'A')

a. Implement classic Caesar encryption, which shifts capital letters accordingly.

b. If a word contains lowercase letters,en should first convert them to uppercase letters.

c. (difficult) Refactor your code and use *scriptblock* variables to get a functional-style version of Caesar encryption.

Task 3

Implement a classic 14-15 puzzle[2] using PowerShell.

[1] The Speech Synthesizer is only available on Windows.

[2] Information about the game can be found at https://de.wikipedia.org/wiki/15-Puzzle