Algorithms and Data Structures

The One Hundred Doors Problem

A famous puzzle, reputedly used as an interview question at Google involves 100 doors. We want to keep track of which are open and which are closed. The problem lends itself to an array, which we'll call doors. The doors are all closed to start with.

A program with a loop is already created for you. It is used to visit each position in the array, setting the value to closed. Note how it does this.

The Problem

You make 100 passes along the line of doors. When you visit a door, if the door is closed, open it; if it is open, close it. On the first pass, visit every door. The second time you only visit every 2nd door. The third time, only every 3rd door … until the 100th pass visits only the 100th door.

Every visit to a door involves toggling it i.e. changing it from open to closed, or vice versa. On the second pass the doors 2, 4, 6, 8 etc. are visited. On the third, it is doors 3, 6, 9, 12 and so on.

Challenge 1: Decomposing the problem

In the box below outline a series of smaller steps to build a working program. The first step is already done, and a second step is given to get you started.

> Create an array (doors) with 100 elements. Iterate over the array setting the value of each element to 'closed'.
>
> Once set, iterate over the array starting at index position 2, changing every second door.

Challenge 2: Write the program

Once you have a plan above, implement each stage in turn. Make sure you test each stage to ensure it is working, before tackling the next one.

Hint: If you are struggling with visiting only certain doors check out the 'step' parameter in a FOR loop.

What is the state of the doors after the last pass. Which are open, which are closed?

Can you see a pattern in the answer?

Challenge 3: Enhance the program

Have you used sub-routines in your solution? It would make sense to encapsulate the code to toggle the door value (open/closed) in a separate procedure. If you haven't, make the necessary enhancements.

Keyword

Step

The Step keyword is used to specify an increment in the For loop.

Example
The following example will print out odd numbers from 1 to 10

```
For i = 1 to 10 Step 2
    TextWindow.WriteLine(i)
EndFor
```