

δ γ % ζ ι  
ρ ε σ π ε δ ε α  
ν υ ζ ρ 8 σ δ ε  
ρ χ ρ υ ο χ σ ρ  
δ φ Ε δ ρ β  
μ 6 β ε τ ε θ σ  
τ ν υ μ κ  
ρ π κ 5 + ε λ β χ ο ε  
η ο μ χ λ β  
υ δ σ ω α ρ φ χ π  
1 β ‡ \* μ ψ λ χ α ζ ε τ

LEVEL 4

Alright, I know how to  
memorize some things now ...

[ yet again, the white rabbit  
hops its way towards a wall of  
green text ... ]

what in the world is a  
postpone instruction?



**New in Level 4:**  
Postpone Statement

In this level a new statement is introduced, which is the **postpone** statement and it is defined as the following:

- **postpone** <statement> \* **end**

### Where

- <statement> \* means zero or more code statements.

The execution of this statement adds the provided statements to the end of the **execution queue**. Which means they will be executed when all other statements queued before them are done.

The **if / else** statement (both parts, if and else) has its own **execution queue**, which means all statements should be done first before any other statement outside.

All statements outside a **postpone** statement are added to the **execution queue** by default in their order of appearance.

Considering these changes execute all the functions like in the previous levels.

	Input	Output
Example 1	<pre> 9 start postpone postpone print Third end print Second end print first end </pre>	<pre> firstSecondThird </pre>
Example 2	<pre> 10 start if false end else postpone print first end end print Second end </pre>	<pre> firstSecond </pre>

	Input	Output
Example 3	<pre>15 start print number if true postpone postpone postpone return 1 end end end end else end print notexecuted end</pre>	<pre>number</pre>



Good  
LUCK!