

Control Structures - While & Until

While and until statements execute a block over and over again until a condition is satisfied. For example:

```
$ dice, rolls.
```

```
dice := 0.
```

```
rolls := 0.
```

```
until [ dice == 7 ] do [
```

```
  dice := (choose 1 to 6) + (choose 1 to 6).
```

```
  rolls := rolls + 1.
```

```
].
```

```
"It took" && rolls.name && "rolls to roll a 7".
```

When the until statement is executed, the expression `dice == 7` is evaluated. If it isn't true, then the block is executed and the whole process repeated. When the condition becomes true, then the until statement finishes.

!!

Note that the condition is in square brackets. Actually, it is a small block. The condition must be in square brackets because it is executed multiple times.

The while statement works the same way, only it keeps executing the block while the condition is true and stops executing when the condition is false. For example:

```
$ n.
```

```
n := 1.
```

```
while [ (n fib) < 100 ] do [
```

```
n := n + 1.
```

```
].
```

```
(n fib).name && "is the first Fibonacci number greater than 100"
```

Infinite Loops

Be careful to ensure that the condition will eventually cause the loop to stop. If you accidentally execute a statement that is never going to end (or if it's just going to take too long), you can interrupt it.

To interrupt a running script:

- 1.

Press the pen down anywhere near the lower left hand corner of the screen. You should press within an inch of the corner. You will have to hold the pen until the User Interrupt note appears. This may take a few seconds.

- 2.

Tap Halt in the User Interrupt note. After a few seconds, a debugger will open. You can simply close it.