**while — while (exp) stmt**

Syntax

```
while (exp) stmt
```

```
while (exp) {
    stmts
}
```

where `exp` must evaluate to a real scalar.

Description

`while` executes `stmt` or `stmts` zero or more times. The loop continues as long as `exp` is not equal to zero.

Remarks and examples

To understand `while`, enter the following program

```
function example(n)
{
    i = 1
    while (i<=n) {
        printf("i=\%g\n", i)
        i++
    }
    printf("done\n")
}
```

and run `example(3)`, `example(2)`, `example(1)`, `example(0)`, and `example(-1)`.

One common use of `while` is to loop until convergence:

```
while (mreldif(a, lasta)>1e-10) {
    lasta = a
    a = ...
}
```
2  while — while (exp) stmt

Also see

[M-2] semicolons — Use of semicolons
[M-2] do — do . . . while (exp)
[M-2] for — for (exp1; exp2; exp3) stmt
[M-2] break — Break out of for, while, or do loop
[M-2] continue — Continue with next iteration of for, while, or do loop
[M-2] intro — Language definition