

# Symbols

Symbols are defined with an equals sign.

```
N = 212^17
```

No result is printed when a symbol is defined. To see the value of a symbol, just evaluate it.

```
N
```

```
N = 3529471145760275132301897342055866171392
```

Symbols can have more than one letter. Everything after the first letter is displayed as a subscript.

```
NA = 6.02214 10^23
```

```
NA
```

```
NA = 6.02214 × 1023
```

A symbol can be the name of a Greek letter.

```
xi = 1/2
```

```
xi
```

```
xi = 1/2
```

Greek letters can appear in subscripts.

```
Amu = 2.0
```

```
Amu
```

```
Aμ = 2.0
```

The following example shows how a symbol is scanned to find Greek letters.

```
alphamunu = 1
```

```
alphamunu
```

```
αμν = 1
```

Symbol definitions are evaluated serially until a terminal symbol is reached. The following example sets  $A = B$  followed by  $B = C$ . Then when  $A$  is evaluated, the result is  $C$ .

```
A = B
```

```
B = C
```

```
A
```

$A = C$

Although  $A = C$  is printed, inside the program the binding of  $A$  is still  $B$ , as can be seen with the `binding` function.

```
binding(A)
```

$B$

The `quote` function returns its argument unevaluated and can be used to clear a symbol. The following example clears  $A$  so that its evaluation goes back to being  $A$  instead of  $C$ .

```
A = quote(A)
```

```
A
```

$A$