

convert between strings and numeric values, and are discussed in detail in Section 4.

Relations

In writing IF commands you use logical formulas, which have a value of true or false. You can build logical formulas by using the arithmetic relations < (less than), > (greater than), = (equal to), <= (less than or equal to), >= (greater than or equal to), and <> (not equal to) between two formulas. You can also combine logical formulas with the logical operators AND, OR (inclusive or), XOR (exclusive or) and NOT (negation), as described in detail below. You can think of the relations and logical operators as arranged in the following order:

<, >, =, <=, >=, <>
NOT
AND
OR, XOR

In addition, arithmetic operations are always performed before logical operations. For example:

X = 1 AND Y = 3 + X * 2 means
(X = 1) AND (Y = (3 + (X * 2)))
PET = DOG AND NOISE = BARK OR PET = CAT AND NOISE = MEOW means
((PET=DOG) AND (NOISE=BARK)) OR ((PET=CAT) AND (NOISE=MEOW))
A * B + C = D AND E < F * G OR H JOIN J * K + L = - M means
((((A*B)+C)=D) AND (E<(F*G))) OR (((H JOIN J)*K)+L)=(-M))

The remainder of this section describes the numeric functions available in XYBASIC. A function is like an operator but is written differently: its arguments (if any) are enclosed in parentheses following the function name. In Extended XYBASIC, a TM (Type Mismatch) error will occur if a string value is used as the argument of a numeric function, and an OV (Overflow) error will occur if a numeric value less than -32767 or greater than 32767 is used where an integer value is expected.

ABS

The ABS function returns the absolute value of its argument. Try some examples:

```
PRINT ABS (-1)
1
OK
PRINT ABS (100)
100
OK
PRINT ABS (-125)
125
OK
```