

The result type of addition, subtraction and multiplication is Integer if both operands are Integer, otherwise it is Real.

Evaluating a term of the form x / y is an error if y is zero.

Evaluating a term of the form $x \text{ div } y$ is an error if y is zero; otherwise the term yields the value satisfying the two rules:

- (a) $\text{abs}(x) - \text{abs}(y) < \text{abs}((x \text{ div } y) * y) \leq \text{abs}(x)$
- (b) $x \text{ div } y = 0$ if $\text{abs}(x) < \text{abs}(y)$, otherwise $x \text{ div } y$ is positive if x and y have the same sign and is negative if x and y have different signs.

Evaluation of a term of the form $x \text{ mod } y$ is an error if y is less than or equal to zero; otherwise there is an integer k such that $x \text{ mod } y$ satisfies the following relation:

$$0 \leq x \text{ mod } y = x - k * y < y.$$

For any integer operators, if both operands are in the range $-\text{Maxint}..\text{Maxint}$ and if the correct result is in that range, then a standard implementation must yield the correct result. However, if the operands or result is not in the range $-\text{Maxint}..\text{Maxint}$, an implementation may choose either to perform the operation correctly or to treat the operation as an error.

Any operator or predeclared function (see Section 11.5) that yields a real result must always be considered to be approximate, not exact. The accuracy of real operations and predeclared functions is implementation-defined.

8.2.2. Boolean Operators. The Boolean operators are summarized by the following table.

<i>Operator</i>	<i>Operation</i>	<i>Type of Operands</i>	<i>Type of Result</i>
or	logical "or"	Boolean	Boolean
and	logical "and"	Boolean	Boolean
not	logical "not"	Boolean	Boolean

8.2.3. Set Operators. The set operators are summarized by the following table. The two operands must always possess compatible types (see Section 6.5). The result type is packed if both operand types are packed, and is non-packed if both operand types are non-packed.

<i>Operator</i>	<i>Operation</i>	<i>Type of Operands</i>	<i>Type of Result</i>
+	set union	set of T	set of T
-	set difference	set of T	set of T
*	set intersection	set of T	set of T