

Lists in MPL

An MPL *list* is a finite collection of expressions that is surrounded by the brackets $[$ and $]$. For example, the expression $[y(x) = 3, x = 1]$ is a list with two equations. The empty list, which contains no expressions, is represented by $[]$.

Lists are distinguished from sets by the following two properties:

1. *The order of expressions in a list is significant.* This means the expressions $[y(x) = 3, x = 1]$ and $[x = 1, y(x) = 3]$ represent different lists.
2. *Duplicate elements are permitted in a list.* This means the expressions $[x, y]$ and $[x, y, y]$ represent different lists.

Primitive Operations on Lists

Let L , M , and N represent lists and let x represent an arbitrary expression. The MPL operations for lists reflect the order preserving property:

- *First(L).* If L contains one or more expressions, the operator returns the first expression in L . If $L = []$, the operator returns the symbol **Undefined**. For example, $First([a, b, c]) \rightarrow a$.
- *Rest(L).* If L contains one or more expressions, the operator returns a new list that contains all expressions in L except the first expression. The original list L is not changed by this operation. If $L = []$, the operator returns the symbol **Undefined**. For example, $Rest([a, b, c]) \rightarrow [b, c]$.
- *Adjoin(x, L).* The operator returns a new list that contains the expression x followed by expressions in L . The original list L is not changed by this operation. For example, $Adjoin(d, [a, b, c]) \rightarrow [d, a, b, c]$.
- *Join(L, M, \dots, N).* The operator returns a new list that contains the expressions in the list L followed by the expressions in M and so on. For example, $Join([a, b], [b, c], [c, d, e]) \rightarrow [a, b, b, c, c, d, e]$.
- *List membership, $x \in L$.* The operator returns **true** if x is in L , and otherwise returns **false**. For example, $b \in [a, b, c] \rightarrow \mathbf{true}$.

Most computer algebra languages provide lists and the list operations described above (see Fig. 1).

MPL	Maple	Mathematica	MuPAD
list notation $[a, b, c]$	$[a, b, c]$	$\{a, b, c\}$	$[a, b, c]$
empty list $[]$	$[]$	$\{\}$	$[]$
<i>First(L)</i>	$\text{op}(1, L)$	$\text{First}[L]$	$L[1]$
<i>Rest(L)</i>	$[\text{op}(2.. \text{nops}(L), L)]$	$\text{Rest}[L]$	$[\text{op}(L, 2.. \text{nops}(L))]$
<i>Adjoin(x, L)</i>	$[x, \text{op}(L)]$	$\text{Prepend}[L, x]$	$\text{append}(L, x)$
<i>Join(L, M)</i>	$[\text{op}(L), \text{op}(M)]$	$\text{Join}[L, M]$	$_ \text{concat}(L, M)$
<i>$x \in L$</i>	$\text{member}(x, L)$	$\text{MemberQ}[x, L]$	$\text{contains}(L, x)$

Figure 1. List operations in Maple, Mathematica, and MuPAD. (Implementation: [Maple](#) (mws), [Mathematica](#) (nb), [MuPAD](#) (mnb).)

[Return to Chapter 1, page 4](#)