
5. ARRAYS

An Interlisp array is a one-dimensional vector of objects. Arrays are generally created by the function `ARRAY`. By contrast, Common Lisp arrays can be multi-dimensional.

Note: Interlisp arrays and Common Lisp arrays are *not* the same types. Interlisp functions only accept Interlisp arrays and vice versa. There are no functions to convert between the two types.

(`ARRAY` *SIZE TYPE INIT ORIG* `-`) [Function]

Creates and returns a new array that holds *SIZE* objects of type *TYPE*. If *TYPE* is `NIL`, the array can contain any arbitrary Lisp datum. In general, *TYPE* may be any of the various field specifications that are legal in `DATATYPE` declarations (see Chapter 8): `POINTER`, `FIXP`, `FLOATP`, `(BITS N)`, etc. Medley will, if necessary, choose an “enclosing” type if the given one is not supported; for example, an array of `(BITS 3)` may be represented by an array of `(BITS 8)`.

INIT is the initial value for each element of the new array. If not specified, the array elements will be initialized with 0 (for number arrays) or `NIL` (all other types).

Arrays can have either 0-origin or 1-origin indexing, as specified by the *ORIG* argument; if *ORIG* is not specified, the default is 1.

Arrays of type `FLOATP` are stored unboxed. This increases the space and time efficiency of `FLOATP` arrays. If you want to use boxed floating point numbers, use an array of type `POINTER` instead of `FLOATP`.

(`ARRAYP` *X*) [Function]

Returns *X* if *X* is an array, `NIL` otherwise.

(`ELT` *ARRAY N*) [Function]

Returns the *N*th element of the array *ARRAY*.

Causes the error, `Arg not array`, if *ARRAY* is not an array. Causes the error, `Illegal Arg`, if *N* is out of bounds.

(`SETA` *ARRAY N VAL*) [Function]

Sets the *N*th element of *ARRAY* to *VAL*, and returns *VAL*.

Causes the error, `Arg not array`, if *ARRAY* is not an array. the error, `Illegal Arg`, if *N* is out of bounds. Can cause the error, `Non-numeric arg`, if *ARRAY* is an array whose `ARRAYTYP` is `FIXP` or `FLOATP` and *VAL* is non-numeric.

(`ARRAYTYP` *ARRAY*) [Function]

Returns the type of the elements in *ARRAY*, a value corresponding to the second argument to *ARRAY*.

INTERLISP-D REFERENCE MANUAL

If `ARRAY` coerced the array type as described above, `ARRAYTYP` returns the *new* type. For example, `(ARRAYTYP (ARRAY 10 ' (BITS 3)))` returns `BYTE`.

(ARRAYSIZE *ARRAY*) [Function]

Returns the size of *ARRAY*. Generates the error, `Arg not array`, if *ARRAY* is not an array.

(ARRAYORIG *ARRAY*) [Function]

Returns the origin of *ARRAY*, which may be 0 or 1. Generates an error, `Arg not array`, if *ARRAY* is not an array.

(COPYARRAY *ARRAY*) [Function]

Returns a new array of the same size and type as *ARRAY*, and with the same contents as *ARRAY*. Generates an error, `Arg not array`, if *ARRAY* is not an array.

ARRAYS

[This page intentionally left blank]