System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main][Doc][c-api]float.rst, line 1)

Unknown directive type "highlight".

```
.. highlight:: c
```

Floating Point Objects

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main][Doc][c-api]float.rst, line 8)

Unknown directive type "index".

```
.. index:: object: floating point
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main][Doc][c-api]float.rst, line 11)

Unknown directive type "c:type".

```
.. c:type:: PyFloatObject

   This subtype of :c:type:`PyObject` represents a Python floating point object.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main][Doc][c-api]float.rst, line 16)

Unknown directive type "c:var".

```
.. c:var:: PyTypeObject PyFloat_Type

   This instance of :c:type:`PyTypeObject` represents the Python floating point
   type.  This is the same object as :class:`float` in the Python layer.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main][Doc][c-api]float.rst, line 22)

Unknown directive type "c:function".

```
.. c:function:: int PyFloat_Check(PyObject *p)

   Return true if its argument is a :c:type:`PyFloatObject` or a subtype of
   :c:type:`PyFloatObject`.  This function always succeeds.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main][Doc][c-api]float.rst, line 28)

Unknown directive type "c:function".

```
.. c:function:: int PyFloat_CheckExact(PyObject *p)

   Return true if its argument is a :c:type:`PyFloatObject`, but not a subtype of
   :c:type:`PyFloatObject`.  This function always succeeds.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main][Doc][c-api]float.rst, line 34)

Unknown directive type "c:function".

```
.. c:function:: PyObject* PyFloat_FromString(PyObject *str)
```

```
   Create a :c:type:`PyFloatObject` object based on the string value in *str*, or
   ``NULL`` on failure.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main][Doc][c-api]float.rst, line 40)

Unknown directive type "c:function".

```
.. c:function:: PyObject* PyFloat_FromDouble(double v)

   Create a :c:type:`PyFloatObject` object from *v*, or ``NULL`` on failure.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main][Doc][c-api]float.rst, line 45)

Unknown directive type "c:function".

```
.. c:function:: double PyFloat_AsDouble(PyObject *pyfloat)

   Return a C :c:type:`double` representation of the contents of *pyfloat*.  If
   *pyfloat* is not a Python floating point object but has a :meth:`__float__`
   method, this method will first be called to convert *pyfloat* into a float.
   If ``__float__()`` is not defined then it falls back to :meth:`__index__`.
   This method returns ``-1.0`` upon failure, so one should call
   :c:func:`PyErr_Occurred` to check for errors.

   .. versionchanged:: 3.8
      Use :meth:`__index__` if available.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main][Doc][c-api]float.rst, line 58)

Unknown directive type "c:function".

```
.. c:function:: double PyFloat_AS_DOUBLE(PyObject *pyfloat)

   Return a C :c:type:`double` representation of the contents of *pyfloat*, but
   without error checking.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main][Doc][c-api]float.rst, line 64)

Unknown directive type "c:function".

```
.. c:function:: PyObject* PyFloat_GetInfo(void)

   Return a structseq instance which contains information about the
   precision, minimum and maximum values of a float. It's a thin wrapper
   around the header file :file:`float.h`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main][Doc][c-api]float.rst, line 71)

Unknown directive type "c:function".

```
.. c:function:: double PyFloat_GetMax()

   Return the maximum representable finite float *DBL_MAX* as C :c:type:`double`.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main][Doc][c-api]float.rst, line 76)

Unknown directive type "c:function".

```
.. c:function:: double PyFloat_GetMin()

   Return the minimum normalized positive float *DBL_MIN* as C :c:type:`double`.
```

Pack and Unpack functions

The pack and unpack functions provide an efficient platform-independent way to store floating-point values as byte strings. The Pack routines produce a bytes string from a C :c:type:`double`, and the Unpack routines produce a C :c:type:`double` from such a bytes string. The suffix (2, 4 or 8) specifies the number of bytes in the bytes string.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main][Doc][c-api]float.rst, line 84); *backlink*

Unknown interpreted text role "c:type".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main][Doc][c-api]float.rst, line 84); *backlink*

Unknown interpreted text role "c:type".

On platforms that appear to use IEEE 754 formats these functions work by copying bits. On other platforms, the 2-byte format is identical to the IEEE 754 binary16 half-precision format, the 4-byte format (32-bit) is identical to the IEEE 754 binary32 single precision format, and the 8-byte format to the IEEE 754 binary64 double precision format, although the packing of INFs and NaNs (if such things exist on the platform) isn't handled correctly, and attempting to unpack a bytes string containing an IEEE INF or NaN will raise an exception.

On non-IEEE platforms with more precision, or larger dynamic range, than IEEE 754 supports, not all values can be packed; on non-IEEE platforms with less precision, or smaller dynamic range, not all values can be unpacked. What happens in such cases is partly accidental (alas).

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main][Doc][c-api]float.rst, line 104)

Unknown directive type "versionadded".

```
.. versionadded:: 3.11
```

Pack functions

The pack routines write 2, 4 or 8 bytes, starting at *p*. *le* is an :c:type:`int` argument, non-zero if you want the bytes string in little-endian format (exponent last, at `p+1`, `p+3`, or `p+6` `p+7`), zero if you want big-endian format (exponent first, at *p*). The :c:data:`PY_BIG_ENDIAN` constant can be used to use the native endian: it is equal to `1` on big endian processor, or `0` on little endian processor.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main][Doc][c-api]float.rst, line 109); *backlink*

Unknown interpreted text role "c:type".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main][Doc][c-api]float.rst, line 109); *backlink*

Unknown interpreted text role "c:data".

Return value: `0` if all is OK, `-1` if error (and an exception is set, most likely :exc:`OverflowError`).

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main][Doc][c-api]float.rst, line 116); *backlink*

Unknown interpreted text role "exc".

There are two problems on non-IEEE platforms:

- What this does is undefined if *x* is a NaN or infinity.
- `-0.0` and `+0.0` produce the same bytes string.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main][Doc][c-api]float.rst, line 124)

Unknown directive type "c:function".

```
.. c:function:: int PyFloat_Pack2(double x, unsigned char *p, int le)

   Pack a C double as the IEEE 754 binary16 half-precision format.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main][Doc][c-api]float.rst, line 128)

Unknown directive type "c:function".

```
.. c:function:: int PyFloat_Pack4(double x, unsigned char *p, int le)

   Pack a C double as the IEEE 754 binary32 single precision format.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main][Doc][c-api]float.rst, line 132)

Unknown directive type "c:function".

```
.. c:function:: int PyFloat_Pack8(double x, unsigned char *p, int le)

   Pack a C double as the IEEE 754 binary64 double precision format.
```

Unpack functions

The unpack routines read 2, 4 or 8 bytes, starting at *p*. *le* is an :c:type:`int` argument, non-zero if the bytes string is in little-endian format (exponent last, at `p+1`, `p+3` or `p+6` and `p+7`), zero if big-endian (exponent first, at *p*). The :c:data:`PY_BIG_ENDIAN` constant can be used to use the native endian: it is equal to `1` on big endian processor, or `0` on little endian processor.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main][Doc][c-api]float.rst, line 140); *backlink*

Unknown interpreted text role "c:type".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main][Doc][c-api]float.rst, line 140); *backlink*

Unknown interpreted text role "c:data".

Return value: The unpacked double. On error, this is `-1.0` and :c:func:`PyErr_Occurred` is true (and an exception is set, most likely :exc:`OverflowError`).

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main][Doc][c-api]float.rst, line 147); *backlink*

Unknown interpreted text role "c:func".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main][Doc][c-api]float.rst, line 147); *backlink*

Unknown interpreted text role "exc".

Note that on a non-IEEE platform this will refuse to unpack a bytes string that represents a NaN or infinity.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main][Doc][c-api]float.rst, line 154)

Unknown directive type "c:function".

```
.. c:function:: double PyFloat_Unpack2(const unsigned char *p, int le)

   Unpack the IEEE 754 binary16 half-precision format as a C double.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main][Doc][c-api]float.rst, line 158)

Unknown directive type "c:function".

```
.. c:function:: double PyFloat_Unpack4(const unsigned char *p, int le)

   Unpack the IEEE 754 binary32 single precision format as a C double.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\c-api\[cpython-main][Doc][c-api]float.rst, line 162)

Unknown directive type "c:function".

```
.. c:function:: double PyFloat_Unpack8(const unsigned char *p, int le)

   Unpack the IEEE 754 binary64 double precision format as a C double.
```