CPSC 189: FINAL EXAMINATION
PYTHON DOCUMENTATION

Note: in the documentation that follows, *iterable* could be a list or tuple. Any arguments in square brackets [] are optional.

abs(*x*)
Return the absolute value of a number. The argument may be a plain or long integer or a floating point number. If the argument is a complex number, its magnitude is returned.

all(*iterable*)
Return True if all elements of the *iterable* are true (or if the iterable is empty).

any(*iterable*)
Return True if any element of the *iterable* is true. If the iterable is empty, return False.

float(*x*)
Convert a string or a number to floating point.

int(*x*)
Convert a number or string *x* to an integer,

len(*iterable*)
Return the length (the number of items) of an *iterable*.

max(*iterable*)
max(*arg1*, *arg2*, **args*)
Return the largest item in an *iterable* or the largest of two or more arguments.

min(*iterable*)
min(*arg1*, *arg2, *args*)
Return the smallest item in an *iterable* or the smallest of two or more arguments.

open(*name, mode*)
Open a file of the given *name*, returning an object of the **file** type; *mode* is one of 'U' for universal read or 'w' for write.

sum(*iterable*[, *start*])
Sums *start* and the items of an *iterable* from left to right and returns the total. *start* defaults to 0.

zip([*iterable*, ...])
This function returns an iterator over tuples, where the *i*-th tuple contains the *i*-th element from each of the argument iterables.

os.path module

os.path.**join**(*path1*[, *path2*[, ...]])
Join one or more path components intelligently.

os.path.**splitext**(*path*)
Split the pathname *path* into a pair (root, ext) such that root + ext == path, and *ext* is empty or begins with a period and contains at most one period.

os module

os.**walk**(*top*)
Generate the file names in a directory tree by walking the tree. For each directory in the tree rooted at directory *top* (including *top* itself), it yields a 3-tuple (dirpath, dirnames, filenames).

dirpath is a string, the path to the directory. *dirnames* is a list of the names of the subdirectories in *dirpath*. *filenames* is a list of the names of the non-directory files in *dirpath*. Note that the names in the lists contain no path components. To get a full path (which begins with *top*) to a file or directory in *dirpath*, do os.path.join(dirpath, name).

String methods

str.**center**(*width*[, *fillchar*])
Return centered in a string of length *width*. Padding is done using the specified *fillchar* (default is a space).

str.**find**(*sub*[, *start*[, *end*]])
Return the lowest index in the string where substring *sub* is found, such that *sub* is contained in the slice s[start:end]. Optional arguments *start* and *end* are interpreted as in slice notation. Return -1 if *sub* is not found.

str.**split**([*sep*])
Return a list of the words in the string, using *sep* as the delimiter string.

If *sep* is given, consecutive delimiters are not grouped together and are deemed to delimit empty strings (for example, '1,,2'.split(',') returns ['1', '','2']). Splitting an empty string with a specified separator returns [''].

If *sep* is not specified or is **None**, a different splitting algorithm is applied: runs of consecutive whitespace are regarded as a single separator, and the result will contain no empty strings at the start or end if the string has leading or trailing whitespace. Consequently, splitting an empty string or a string consisting of just whitespace with a **None** separator returns [].
For example, ' 1 2   3 '.split() returns ['1', '2', '3'].

str.**strip**()
Return a copy of the string with the leading and trailing characters removed.

numpy library

Attributes of ndarray objects:

ndarray.**ndim**
Number of array dimensions.

ndarray.**size**
Number of elements in the array.

ndarray.**shape**
Tuple of array dimensions.

Functions of the numpy library:

numpy.**min**(*ndarray*, *axis=None*)
Return the minimum of *ndarray* along the given axis or across the entire array if axis is not specified.

numpy.**max**(*ndarray*, *axis=None*)
Return the maximum of *ndarray* along the given axis or across the entire array if axis is not specified.

numpy.**sum**(*ndarray*, *axis=None*)
Return the sum of *ndarray* along the given axis or across the entire array if axis is not specified.

numpy.**mean**(*ndarray, axis=None*)
Returns the average of *ndarray* along the given axis or across the entire array if axis is not specified.

numpy.**all**(*ndarray, axis=None*)
Determines whether all elements along the given axis are True. Determines whether all elements across the entire array are True if axis is not specified.

numpy.**any**(*ndarray*, *axis=None*)
Determines whether any element along the given axis is True. Determines whether any element across the entire array is True if axis is not specified.