Functools

@cache – function decorator (goes on the line before function definition) that stores input/output pairs so if an input is received a second time, it can simply look up the output. Good for when repetitive math takes too long such as factorials and the Fibonacci sequence.

@lru\_cache(n) – very similar to cache, however it only stores the n most recent input/output pairs. Good for situations where recent inputs are likely to be used again soon.

cmp\_to\_key(func) – converts a C-like comparison function (a function that returns -1, 0, or 1 for comparisons) to a Python style key function. Useful in sorting because Python doesn’t directly support comparison functions.

reduce(func, iterable) – similar to itertools.accumulate(), it applies func over iterable until all values have been used, then returns the result. Good for getting the product, sum, or other result with all values in a list.