

Code

```
1  #!/usr/bin/env python
2  # -*- coding: utf-8 -*-
3  # __coconut_hash__ = 0xaaa2e80a
4
5  # Compiled with Coconut version 1.4.3 [Ernest Scribbler]
6
7  # Coconut Header: -----
8
9  from __future__ import print_function, absolute_import, unicode_literals, division
10 import sys as _coconut_sys
11 if _coconut_sys.version_info < (3,):
12     from __builtin__ import chr, filter, hex, input, int, map, object, oct, open,
13     py_chr, py_hex, py_input, py_int, py_map, py_object, py_oct, py_open, py_print
14     _coconut_NotImplemented, _coconut_raw_input, _coconut_xrange, _coconut_int, _coconut_long
15     from future_builtins import *
16     chr, str = unichr, unicode
17     from io import open
18     class object(object):
19         __slots__ = ()
20         def __ne__(self, other):
21             eq = self == other
22             if eq is _coconut_NotImplemented:
23                 return eq
24             return not eq
25     class int(_coconut_int):
26         __slots__ = ()
27         if hasattr(_coconut_int, "__doc__"):
28             __doc__ = _coconut_int.__doc__
29         class __metaclass__(type):
30             def __instancecheck__(cls, inst):
31                 return _coconut.isinstance(inst, (_coconut_int, _coconut_long))
32             def __subclasscheck__(cls, subcls):
33                 return _coconut.issubclass(subcls, (_coconut_int, _coconut_long))
34     class range(object):
35         __slots__ = ("xrange",)
36         if hasattr(_coconut_xrange, "__doc__"):
```

```

37     __doc__ = _coconut_xrange.__doc__
38     def __init__(self, *args):
39         self._xrange = _coconut_xrange(*args)
40     def __iter__(self):
41         return _coconut.iter(self._xrange)
42     def __reversed__(self):
43         return _coconut.reversed(self._xrange)
44     def __len__(self):
45         return _coconut.len(self._xrange)
46     def __contains__(self, elem):
47         return elem in self._xrange
48     def __getitem__(self, index):
49         if _coconut.isinstance(index, _coconut.slice):
50             args = _coconut.slice(*self._args)
51             start, stop, step, ind_step = (args.start if args.start is not None else 0,
52                                           args.stop if args.stop is not None else None,
53                                           args.step if args.step is not None else 1,
54                                           1 if args.step >= 0 else -1)
55             return self.__class__((start if ind_step >= 0 else stop - step) if ind_step != 0 else None,
56                                   stop if ind_step >= 0 else start,
57                                   step)
58         else:
59             return self._xrange[index]
60     def count(self, elem):
61         """Count the number of times elem appears in the range."""
62         return _coconut_int(elem in self._xrange)
63     def index(self, elem):
64         """Find the index of elem in the range."""
65         if elem not in self._xrange: raise _coconut.ValueError(_coconut.repr(elem) +
66                                                                   " is not in range")
67         start, _, step = self._xrange.__reduce_ex__(2)[1]
68         return (elem - start) // step
69     def __repr__(self):
70         return _coconut.repr(self._xrange)[1:]
71     @property
72     def _args(self):
73         return self._xrange.__reduce__()[1]
74     def __reduce_ex__(self, protocol):
75         return (self.__class__, self._xrange.__reduce_ex__(protocol)[1])
76     def __reduce__(self):
77         return self.__reduce_ex__(_coconut.pickle.DEFAULT_PROTOCOL)
78     def __hash__(self):
79         return _coconut.hash(self._args)
80     def __copy__(self):

```

```

75         return self.__class__(*self._args)
76     def __eq__(self, other):
77         return _coconut.isinstance(other, self.__class__) and self._args == other._args
78 from collections import Sequence as _coconut_Sequence
79 _coconut_Sequence.register(range)
80 from functools import wraps as _coconut_wraps
81 @_coconut_wraps(_coconut_print)
82 def print(*args, **kwargs):
83     file = kwargs.get("file", _coconut_sys.stdout)
84     flush = kwargs.get("flush", False)
85     if "flush" in kwargs:
86         del kwargs["flush"]
87     if _coconut.hasattr(file, "encoding") and file.encoding is not None:
88         _coconut_print(*(_coconut_unicode(x).encode(file.encoding) for x in args), **kwargs)
89     else:
90         _coconut_print(*(_coconut_unicode(x).encode() for x in args), **kwargs)
91     if flush:
92         file.flush()
93 @_coconut_wraps(_coconut_raw_input)
94 def input(*args, **kwargs):
95     if _coconut.hasattr(_coconut_sys.stdout, "encoding") and _coconut_sys.stdout.encoding is not None:
96         return _coconut_raw_input(*args, **kwargs).decode(_coconut_sys.stdout.encoding)
97     return _coconut_raw_input(*args, **kwargs).decode()
98 @_coconut_wraps(_coconut_repr)
99 def repr(obj):
100     if isinstance(obj, _coconut_unicode):
101         return _coconut_unicode(_coconut_repr(obj)[1:])
102     if isinstance(obj, _coconut_str):
103         return "b" + _coconut_unicode(_coconut_repr(obj))
104     return _coconut_unicode(_coconut_repr(obj))
105 ascii = repr
106 def raw_input(*args):
107     """Coconut uses Python 3 "input" instead of Python 2 "raw_input"."""
108     raise _coconut.NameError('Coconut uses Python 3 "input" instead of Python 2 "raw_input"')
109 def xrange(*args):
110     """Coconut uses Python 3 "range" instead of Python 2 "xrange"."""
111     raise _coconut.NameError('Coconut uses Python 3 "range" instead of Python 2 "xrange"')
112 if _coconut_sys.version_info < (2, 7):

```

```

113         import functools as _coconut_functools, copy_reg as _coconut_copy_reg
114         def _coconut_new_partial(func, args, keywords):
115             return _coconut_functools.partial(func, *(args if args is not None else []), **keywords)
116         _coconut_copy_reg.constructor(_coconut_new_partial)
117         def _coconut_reduce_partial(self):
118             return (_coconut_new_partial, (self.func, self.args, self.keywords))
119         _coconut_copy_reg.pickle(_coconut_functools.partial, _coconut_reduce_partial)
120     else:
121         from builtins import chr, filter, hex, input, int, map, object, oct, open, print, range, repr, sorted,
122         py_chr, py_hex, py_input, py_int, py_map, py_object, py_oct, py_open, py_print, py_range, py_sorted, py_repr,
123         _coconut_str = str
124     class _coconut(object):
125         import collections, copy, functools, types, itertools, operator, threading, weakref
126         if _coconut_sys.version_info < (3, 2):
127             try:
128                 from backports.functools_lru_cache import lru_cache
129                 functools.lru_cache = lru_cache
130             except ImportError: pass
131         if _coconut_sys.version_info < (3,):
132             import cPickle as pickle
133         else:
134             import pickle
135         if _coconut_sys.version_info >= (2, 7):
136             OrderedDict = collections.OrderedDict
137         else:
138             OrderedDict = dict
139         if _coconut_sys.version_info < (3, 3):
140             abc = collections
141         else:
142             import collections.abc as abc
143         class typing(object):
144             @staticmethod
145             def NamedTuple(name, fields):
146                 return _coconut.collections.namedtuple(name, [x for x, t in fields])
147         Ellipsis, Exception, AttributeError, ImportError, IndexError, KeyError, NameError, NotImplementedError,
148         _coconut_sentinel = _coconut.object()
149         class MatchError(Exception):
150             """Pattern-matching error. Has attributes .pattern and .value."""

```

```

151     __slots__ = ("pattern", "value")
152 class _coconut_tail_call(object):
153     __slots__ = ("func", "args", "kwargs")
154     def __init__(self, func, *args, **kwargs):
155         self.func, self.args, self.kwargs = func, args, kwargs
156 _coconut_tco_func_dict = {}
157 def _coconut_tco(func):
158     @_coconut.functools.wraps(func)
159     def tail_call_optimized_func(*args, **kwargs):
160         call_func = func
161         while True:
162             wkref = _coconut_tco_func_dict.get(_coconut.id(call_func))
163             if (wkref is not None and wkref() is call_func) or _coconut.isinstance(
164                 call_func, _coconut_tco_func):
165                 result = call_func(*args, **kwargs) # pass --no-tco to clean up your
166                 if not isinstance(result, _coconut_tail_call):
167                     return result
168                 call_func, args, kwargs = result.func, result.args, result.kwargs
169 tail_call_optimized_func._coconut_tco_func = func
170 tail_call_optimized_func.__module__ = _coconut.getattr(func, "__module__", None)
171 tail_call_optimized_func.__name__ = _coconut.getattr(func, "__name__", "<coconut")
172 tail_call_optimized_func.__qualname__ = _coconut.getattr(func, "__qualname__",
173 _coconut_tco_func_dict[_coconut.id(tail_call_optimized_func)] = _coconut.weakref(
174 return tail_call_optimized_func
175 def _coconut_igetitem(iterable, index):
176     if isinstance(iterable, (_coconut_reversed, _coconut_map, _coconut.zip, _coconut
177         return iterable[index]
178     if not _coconut.isinstance(index, _coconut.slice):
179         if index < 0:
180             return _coconut.collections.deque(iterable, maxlen=-index)[0]
181             return _coconut.next(_coconut.itertools.islice(iterable, index, index + 1))
182     if index.start is not None and index.start < 0 and (index.stop is None or index
183         queue = _coconut.collections.deque(iterable, maxlen=-index.start)
184         if index.stop is not None:
185             queue = _coconut.list(queue)[:index.stop - index.start]
186         return queue
187     if (index.start is not None and index.start < 0) or (index.stop is not None and
188         return _coconut.list(iterable)[index]

```

```

189     return _coconut.itertools.islice(iterable, index.start, index.stop, index.step)
190 class _coconut_base_compose(object):
191     __slots__ = ("func", "funcstars")
192     def __init__(self, func, *funcstars):
193         self.func = func
194         self.funcstars = []
195         for f, stars in funcstars:
196             if _coconut.isinstance(f, _coconut_base_compose):
197                 self.funcstars.append((f.func, stars))
198                 self.funcstars += f.funcstars
199             else:
200                 self.funcstars.append((f, stars))
201     def __call__(self, *args, **kwargs):
202         arg = self.func(*args, **kwargs)
203         for f, stars in self.funcstars:
204             if stars == 0:
205                 arg = f(arg)
206             elif stars == 1:
207                 arg = f(*arg)
208             elif stars == 2:
209                 arg = f(**arg)
210             else:
211                 raise _coconut.ValueError("invalid arguments to " + _coconut.repr(
212                     self))
213     def __repr__(self):
214         return _coconut.repr(self.func) + " " + " ".join(("..*> " if star == 1 else
215             "" for star, _ in self.funcstars))
216     def __reduce__(self):
217         return (self.__class__, (self.func,) + _coconut.tuple(self.funcstars))
218     def __get__(self, obj, objtype=None):
219         return _coconut.functools.partial(self, obj)
220 def _coconut_forward_compose(func, *funcs): return _coconut_base_compose(func, *(
221     _coconut_reverse_compose(*funcs)
222     _coconut_forward_star_compose(func, *funcs): return _coconut_base_compose(func,
223     _coconut_back_star_compose(*funcs): return _coconut_forward_star_compose(*co
224     _coconut_forward_dubstar_compose(func, *funcs): return _coconut_base_compose(
225     _coconut_back_dubstar_compose(*funcs): return _coconut_forward_dubstar_compose
226     _coconut_pipe(x, f): return f(x)
227     _coconut_star_pipe(xs, f): return f(*xs)

```

```

227 def _coconut_dubstar_pipe(kws, f): return f(**kws)
228 def _coconut_back_pipe(f, x): return f(x)
229 def _coconut_back_star_pipe(f, xs): return f(*xs)
230 def _coconut_back_dubstar_pipe(f, kws): return f(**kws)
231 def _coconut_assert(cond, msg=None): assert cond, msg if msg is not None else "(a
232 def _coconut_bool_and(a, b): return a and b
233 def _coconut_bool_or(a, b): return a or b
234 def _coconut_none_coalesce(a, b): return a if a is not None else b
235 def _coconut_minus(a, *rest):
236     if not rest:
237         return -a
238     for b in rest:
239         a = a - b
240     return a
241 @_coconut.functools.wraps(_coconut.itertools.tee)
242 def tee(iterable, n=2):
243     if n >= 0 and _coconut.isinstance(iterable, (_coconut.tuple, _coconut.frozenset)):
244         return (iterable,) * n
245     if n > 0 and (_coconut.hasattr(iterable, "__copy__") or _coconut.isinstance(iterable, (list, dict))):
246         return (iterable,) + _coconut.tuple(_coconut.copy.copy(iterable) for _ in range(n-1))
247     return _coconut.itertools.tee(iterable, n)
248 class reiterable(object):
249     """Allows an iterator to be iterated over multiple times."""
250     __slots__ = ("iter",)
251     def __init__(self, iterable):
252         self.iter = iterable
253     def _get_new_iter(self):
254         self.iter, new_iter = _coconut_tee(self.iter)
255         return new_iter
256     def __iter__(self):
257         return _coconut.iter(self._get_new_iter())
258     def __getitem__(self, index):
259         return _coconut_igetitem(self._get_new_iter(), index)
260     def __reversed__(self):
261         return _coconut_reversed(self._get_new_iter())
262     def __len__(self):
263         return _coconut.len(self.iter)
264     def __repr__(self):

```

```

265         return "reiterable(%r)" % (self.iter,)
266     def __reduce__(self):
267         return (self.__class__, (self.iter,))
268     def __copy__(self):
269         return self.__class__(self._get_new_iter())
270     def __fmap__(self, func):
271         return _coconut_map(func, self)
272 class scan(object):
273     """Reduce func over iterable, yielding intermediate results,
274     optionally starting from initializer."""
275     __slots__ = ("func", "iter", "initializer")
276     def __init__(self, function, iterable, initializer=_coconut_sentinel):
277         self.func = function
278         self.iter = iterable
279         self.initializer = initializer
280     def __iter__(self):
281         acc = self.initializer
282         if acc is not _coconut_sentinel:
283             yield acc
284         for item in self.iter:
285             if acc is _coconut_sentinel:
286                 acc = item
287             else:
288                 acc = self.func(acc, item)
289             yield acc
290     def __len__(self):
291         return _coconut.len(self.iter)
292     def __repr__(self):
293         return "scan(%r, %r)" % (self.func, self.iter)
294     def __reduce__(self):
295         return (self.__class__, (self.func, self.iter))
296     def __copy__(self):
297         return self.__class__(self.func, _coconut.copy.copy(self.iter))
298     def __fmap__(self, func):
299         return _coconut_map(func, self)
300 class reversed(object):
301     __slots__ = ("iter",)
302     if hasattr(_coconut.map, "__doc__"):

```



```

303     __doc__ = _coconut.reversed.__doc__
304 def __new__(cls, iterable):
305     if _coconut.isinstance(iterable, _coconut.range):
306         return iterable[::-1]
307     if not _coconut.hasattr(iterable, "__reversed__") or _coconut.isinstance(
308         iterable, _coconut.object.__new__(cls)
309     ):
310         return _coconut.reversed(iterable)
311 def __init__(self, iterable):
312     self.iter = iterable
313 def __iter__(self):
314     return _coconut.iter(_coconut.reversed(self.iter))
315 def __getitem__(self, index):
316     if _coconut.isinstance(index, _coconut.slice):
317         return _coconut_igetitem(self.iter, _coconut.slice(-(index.start + 1),
318             -(index.stop + 1)))
319     return _coconut_igetitem(self.iter, -(index + 1))
320 def __reversed__(self):
321     return self.iter
322 def __len__(self):
323     return _coconut.len(self.iter)
324 def __repr__(self):
325     return "reversed(%r)" % (self.iter,)
326 def __hash__(self):
327     return -_coconut.hash(self.iter)
328 def __reduce__(self):
329     return (self.__class__, (self.iter,))
330 def __copy__(self):
331     return self.__class__(_coconut.copy.copy(self.iter))
332 def __eq__(self, other):
333     return isinstance(other, self.__class__) and self.iter == other.iter
334 def __contains__(self, elem):
335     return elem in self.iter
336 def count(self, elem):
337     """Count the number of times elem appears in the reversed iterator."""
338     return self.iter.count(elem)
339 def index(self, elem):
340     """Find the index of elem in the reversed iterator."""
341     return _coconut.len(self.iter) - self.iter.index(elem) - 1
342 def __fmap__(self, func):

```

```

341         return self.__class__(_coconut_map(func, self.iter))
342 class map(_coconut.map):
343     __slots__ = ("func", "iters")
344     if hasattr(_coconut.map, "__doc__"):
345         __doc__ = _coconut.map.__doc__
346     def __new__(cls, function, *iterables):
347         new_map = _coconut.map.__new__(cls, function, *iterables)
348         new_map.func = function
349         new_map.iters = iterables
350         return new_map
351     def __getitem__(self, index):
352         if _coconut.isinstance(index, _coconut.slice):
353             return self.__class__(self.func, *(_coconut_igetitem(i, index) for i in self.iters))
354         return self.func(*(_coconut_igetitem(i, index) for i in self.iters))
355     def __reversed__(self):
356         return self.__class__(self.func, *(_coconut_reversed(i) for i in self.iters))
357     def __len__(self):
358         return _coconut.min(_coconut.len(i) for i in self.iters)
359     def __repr__(self):
360         return "map(%r, %s)" % (self.func, ", ".join((_coconut.repr(i) for i in self.iters)))
361     def __reduce__(self):
362         return (self.__class__, (self.func,) + self.iters)
363     def __reduce_ex__(self, _):
364         return self.__reduce__()
365     def __copy__(self):
366         return self.__class__(self.func, *_coconut.map(_coconut.copy.copy, self.iters))
367     def __fmap__(self, func):
368         return self.__class__(_coconut_forward_compose(self.func, func), *self.iters)
369 class parallel_map(map):
370     """Multi-process implementation of map using concurrent.futures.
371     Requires arguments to be pickleable."""
372     __slots__ = ()
373     def __iter__(self):
374         from concurrent.futures import ProcessPoolExecutor
375         with ProcessPoolExecutor() as executor:
376             return _coconut.iter(_coconut.list(executor.map(self.func, *self.iters)))
377     def __repr__(self):
378         return "parallel_" + _coconut_map.__repr__(self)

```

```

379 class concurrent_map(map):
380     """Multi-thread implementation of map using concurrent.futures."""
381     __slots__ = ()
382     def __iter__(self):
383         from concurrent.futures import ThreadPoolExecutor
384         from multiprocessing import cpu_count # cpu_count() * 5 is the default P
385         with ThreadPoolExecutor(cpu_count() * 5) as executor:
386             return _coconut.iter(_coconut.list(executor.map(self.func, *self.iters
387     def __repr__(self):
388         return "concurrent_" + _coconut_map.__repr__(self)
389 class filter(_coconut.filter):
390     __slots__ = ("func", "iter")
391     if hasattr(_coconut.filter, "__doc__"):
392         __doc__ = _coconut.filter.__doc__
393     def __new__(cls, function, iterable):
394         new_filter = _coconut.filter.__new__(cls, function, iterable)
395         new_filter.func = function
396         new_filter.iter = iterable
397         return new_filter
398     def __reversed__(self):
399         return self.__class__(self.func, _coconut_reversed(self.iter))
400     def __repr__(self):
401         return "filter(%r, %r)" % (self.func, self.iter)
402     def __reduce__(self):
403         return (self.__class__, (self.func, self.iter))
404     def __reduce_ex__(self, _):
405         return self.__reduce__()
406     def __copy__(self):
407         return self.__class__(self.func, _coconut.copy.copy(self.iter))
408     def __fmap__(self, func):
409         return _coconut_map(func, self)
410 class zip(_coconut.zip):
411     __slots__ = ("iters",)
412     if hasattr(_coconut.zip, "__doc__"):
413         __doc__ = _coconut.zip.__doc__
414     def __new__(cls, *iterables):
415         new_zip = _coconut.zip.__new__(cls, *iterables)
416         new_zip.iters = iterables

```

```

417         return new_zip
418     def __getitem__(self, index):
419         if _coconut.isinstance(index, _coconut.slice):
420             return self.__class__(*(_coconut_igetitem(i, index) for i in self.iter))
421         return _coconut.tuple(_coconut_igetitem(i, index) for i in self.iters)
422     def __reversed__(self):
423         return self.__class__(*(_coconut_reversed(i) for i in self.iters))
424     def __len__(self):
425         return _coconut.min(_coconut.len(i) for i in self.iters)
426     def __repr__(self):
427         return "zip(%s)" % (" , ".join((_coconut.repr(i) for i in self.iters)),)
428     def __reduce__(self):
429         return (self.__class__, self.iters)
430     def __reduce_ex__(self, _):
431         return self.__reduce__()
432     def __copy__(self):
433         return self.__class__(*_coconut.map(_coconut.copy.copy, self.iters))
434     def __fmap__(self, func):
435         return _coconut_map(func, self)
436 class enumerate(_coconut.enumerate):
437     __slots__ = ("iter", "start")
438     if hasattr(_coconut.enumerate, "__doc__"):
439         __doc__ = _coconut.enumerate.__doc__
440     def __new__(cls, iterable, start=0):
441         new_enumerate = _coconut.enumerate.__new__(cls, iterable, start)
442         new_enumerate.iter = iterable
443         new_enumerate.start = start
444         return new_enumerate
445     def __getitem__(self, index):
446         if _coconut.isinstance(index, _coconut.slice):
447             return self.__class__(_coconut_igetitem(self.iter, index), self.start)
448         return (self.start + index, _coconut_igetitem(self.iter, index))
449     def __len__(self):
450         return _coconut.len(self.iter)
451     def __repr__(self):
452         return "enumerate(%r, %r)" % (self.iter, self.start)
453     def __reduce__(self):
454         return (self.__class__, (self.iter, self.start))

```

```

455     def __reduce_ex__(self, _):
456         return self.__reduce__()
457     def __copy__(self):
458         return self.__class__(_coconut.copy.copy(self.iter), self.start)
459     def __fmap__(self, func):
460         return _coconut_map(func, self)
461 class count(object):
462     """count(start, step) returns an infinite iterator starting at start and inc
463     If step is set to 0, count will infinitely repeat its first argument."""
464     __slots__ = ("start", "step")
465     def __init__(self, start=0, step=1):
466         self.start = start
467         self.step = step
468     def __iter__(self):
469         while True:
470             yield self.start
471             if self.step:
472                 self.start += self.step
473     def __contains__(self, elem):
474         if not self.step:
475             return elem == self.start
476         if elem < self.start:
477             return False
478         return (elem - self.start) % self.step == 0
479     def __getitem__(self, index):
480         if _coconut.isinstance(index, _coconut.slice) and (index.start is None or
481         new_start, new_step = self.start, self.step
482         if self.step and index.start is not None:
483             new_start += self.step * index.start
484         if self.step and index.step is not None:
485             new_step *= index.step
486         if index.stop is None:
487             return self.__class__(new_start, new_step)
488         if self.step and _coconut.isinstance(self.start, _coconut.int) and _co
489             return _coconut.range(new_start, self.start + self.step * index.st
490         return _coconut_map(self.__getitem__, _coconut.range(index.start if i
491 if index < 0:
492     raise _coconut.IndexError("count indices must be positive")

```

```

493         return self.start + self.step * index if self.step else self.start
494     def count(self, elem):
495         """Count the number of times elem appears in the count."""
496         if not self.step:
497             return _coconut.float("inf") if elem == self.start else 0
498         return int(elem in self)
499     def index(self, elem):
500         """Find the index of elem in the count."""
501         if elem not in self:
502             raise _coconut.ValueError(_coconut.repr(elem) + " not in " + _coconut
503             return (elem - self.start) // self.step if self.step else 0
504     def __reversed__(self):
505         if not self.step:
506             return self
507         raise _coconut.TypeError(repr(self) + " object is not reversible")
508     def __repr__(self):
509         return "count(%r, %r)" % (self.start, self.step)
510     def __hash__(self):
511         return _coconut.hash((self.start, self.step))
512     def __reduce__(self):
513         return (self.__class__, (self.start, self.step))
514     def __copy__(self):
515         return self.__class__(self.start, self.step)
516     def __eq__(self, other):
517         return isinstance(other, self.__class__) and self.start == other.start and
518     def __fmap__(self, func):
519         return _coconut_map(func, self)
520 class groupsof(object):
521     """groupsof(n, iterable) splits iterable into groups of size n.
522     If the length of the iterable is not divisible by n, the last group may be o
523     __slots__ = ("group_size", "iter")
524     def __init__(self, n, iterable):
525         self.iter = iterable
526         try:
527             self.group_size = _coconut.int(n)
528         except _coconut.ValueError:
529             raise _coconut.TypeError("group size must be an int; not %r" % (n,))
530         if self.group_size <= 0:

```

```

531         raise _coconut.ValueError("group size must be > 0; not %r" % (self.gr
532 def __iter__(self):
533     iterator = _coconut.iter(self.iter)
534     loop = True
535     while loop:
536         group = []
537         for _ in _coconut.range(self.group_size):
538             try:
539                 group.append(_coconut.next(iterator))
540             except _coconut.StopIteration:
541                 loop = False
542                 break
543         if group:
544             yield _coconut.tuple(group)
545 def __len__(self):
546     return _coconut.len(self.iter)
547 def __repr__(self):
548     return "groupsof(%r)" % (self.iter,)
549 def __reduce__(self):
550     return (self.__class__, (self.group_size, self.iter))
551 def __copy__(self):
552     return self.__class__(self.group_size, _coconut.copy.copy(self.iter))
553 def __fmap__(self, func):
554     return _coconut_map(func, self)
555 class recursive_iterator(object):
556     """Decorator that optimizes a function for iterator recursion."""
557     __slots__ = ("func", "tee_store", "backup_tee_store")
558     def __init__(self, func):
559         self.func = func
560         self.tee_store = {}
561         self.backup_tee_store = []
562     def __call__(self, *args, **kwargs):
563         key = (args, _coconut.frozenset(kwargs))
564         use_backup = False
565         try:
566             hash(key)
567         except _coconut.Exception:
568             try:

```

```

569         key = _coconut.pickle.dumps(key, -1)
570     except _coconut.Exception:
571         use_backup = True
572     if use_backup:
573         for i, (k, v) in _coconut.enumerate(self.backup_tee_store):
574             if k == key:
575                 to_tee, store_pos = v, i
576                 break
577             else: # no break
578                 to_tee = self.func(*args, **kwargs)
579                 store_pos = None
580         to_store, to_return = _coconut_tee(to_tee)
581         if store_pos is None:
582             self.backup_tee_store.append([key, to_store])
583         else:
584             self.backup_tee_store[store_pos][1] = to_store
585     else:
586         self.tee_store[key], to_return = _coconut_tee(self.tee_store.get(key))
587     return to_return
588 def __repr__(self):
589     return "@recursive_iterator(" + _coconut.repr(self.func) + ")"
590 def __reduce__(self):
591     return (self.__class__, (self.func,))
592 def __get__(self, obj, objtype=None):
593     return _coconut.functools.partial(self, obj)
594 class _coconut_FunctionMatchErrorContext(object):
595     __slots__ = ('exc_class', 'taken')
596     threadlocal_var = _coconut.threading.local()
597     def __init__(self, exc_class):
598         self.exc_class = exc_class
599         self.taken = False
600     def __enter__(self):
601         try:
602             self.threadlocal_var.contexts.append(self)
603         except _coconut.AttributeError:
604             self.threadlocal_var.contexts = [self]
605     def __exit__(self, type, value, traceback):
606         self.threadlocal_var.contexts.pop()

```



```

607     @classmethod
608     def get(cls):
609         try:
610             ctx = cls.threadlocal_var.contexts[-1]
611         except (_coconut.AttributeError, _coconut.IndexError):
612             return _coconut_MatchError
613         if not ctx.taken:
614             ctx.taken = True
615             return ctx.exc_class
616         return _coconut_MatchError
617 _coconut_get_function_match_error = _coconut_FunctionMatchErrorContext.get
618 class _coconut_base_pattern_func(object):
619     __slots__ = ("FunctionMatchError", "__doc__", "patterns")
620     _coconut_is_match = True
621     def __init__(self, *funcs):
622         self.FunctionMatchError = _coconut.type(_coconut_str("MatchError"), (_coconut,))
623         self.__doc__ = None
624         self.patterns = []
625         for func in funcs:
626             self.add(func)
627     def add(self, func):
628         self.__doc__ = _coconut.getattr(func, "__doc__", None) or self.__doc__
629         if _coconut.isinstance(func, _coconut_base_pattern_func):
630             self.patterns += func.patterns
631         else:
632             self.patterns.append(func)
633     def __call__(self, *args, **kwargs):
634         for func in self.patterns[:-1]:
635             try:
636                 with _coconut_FunctionMatchErrorContext(self.FunctionMatchError):
637                     return func(*args, **kwargs)
638             except self.FunctionMatchError:
639                 pass
640         return self.patterns[-1](*args, **kwargs)
641     def _coconut_tco_func(self, *args, **kwargs):
642         for func in self.patterns[:-1]:
643             try:
644                 with _coconut_FunctionMatchErrorContext(self.FunctionMatchError):

```

```

645         return func(*args, **kwargs)
646     except self.FunctionMatchError:
647         pass
648     return _coconut_tail_call(self.patterns[-1], *args, **kwargs)
649 def __repr__(self):
650     return "addpattern(" + _coconut.repr(self.patterns[0]) + ")(*" + _coconut
651 def __reduce__(self):
652     return (self.__class__, _coconut.tuple(self.patterns))
653 def __get__(self, obj, objtype=None):
654     if obj is None:
655         return self
656     return _coconut functools.partial(self, obj)
657 def _coconut_mark_as_match(base_func):
658     base_func._coconut_is_match = True
659     return base_func
660 def addpattern(base_func, **kwargs):
661     """Decorator to add a new case to a pattern-matching function,
662     where the new case is checked last."""
663     allow_any_func = kwargs.pop("allow_any_func", False)
664     if not allow_any_func and not _coconut.getattr(base_func, "_coconut_is_match"):
665         _coconut.warnings.warn("Possible misuse of addpattern with non-pattern-mat
666     if kwargs:
667         raise _coconut.TypeError("addpattern() got unexpected keyword arguments ")
668     return _coconut functools.partial(_coconut_base_pattern_func, base_func)
669 _coconut_addpattern = addpattern
670 class _coconut_partial(object):
671     __slots__ = ("func", "_argdict", "_arglen", "_stargs", "keywords")
672     if hasattr(_coconut functools.partial, "__doc__"):
673         __doc__ = _coconut functools.partial.__doc__
674     def __init__(self, func, argdict, arglen, *args, **kwargs):
675         self.func = func
676         self._argdict = argdict
677         self._arglen = arglen
678         self._stargs = args
679         self.keywords = kwargs
680     def __reduce__(self):
681         return (self.__class__, (self.func, self._argdict, self._arglen) + self._s
682     def __setstate__(self, keywords):

```

```

683         self.keywords = keywords
684     @property
685     def args(self):
686         return _coconut.tuple(self._argdict.get(i) for i in _coconut.range(self._arglen))
687     def __call__(self, *args, **kwargs):
688         callargs = []
689         argind = 0
690         for i in _coconut.range(self._arglen):
691             if i in self._argdict:
692                 callargs.append(self._argdict[i])
693             elif argind >= _coconut.len(args):
694                 raise _coconut.TypeError("expected at least " + _coconut.str(self._argdict[i]))
695             else:
696                 callargs.append(args[argind])
697                 argind += 1
698         callargs += self._stargs
699         callargs += args[argind:]
700         kwargs.update(self.keywords)
701         return self.func(*callargs, **kwargs)
702     def __repr__(self):
703         args = []
704         for i in _coconut.range(self._arglen):
705             if i in self._argdict:
706                 args.append(_coconut.repr(self._argdict[i]))
707             else:
708                 args.append("?")
709         for arg in self._stargs:
710             args.append(_coconut.repr(arg))
711         return _coconut.repr(self.func) + "$(" + ", ".join(args) + ")"
712     def consume(iterable, keep_last=0):
713         """consume(iterable, keep_last) fully exhausts iterable and return the last keep_last elements"""
714         return _coconut.collections.deque(iterable, maxlen=keep_last)
715     class starmap(_coconut.itertools.starmap):
716         __slots__ = ("func", "iter")
717         if hasattr(_coconut.itertools.starmap, "__doc__"):
718             __doc__ = _coconut.itertools.starmap.__doc__
719         def __new__(cls, function, iterable):
720             new_map = _coconut.itertools.starmap.__new__(cls, function, iterable)

```

```

721         new_map.func = function
722         new_map.iter = iterable
723         return new_map
724     def __getitem__(self, index):
725         if _coconut.isinstance(index, _coconut.slice):
726             return self.__class__(self.func, _coconut_igetitem(self.iter, index))
727         return self.func(*_coconut_igetitem(self.iter, index))
728     def __reversed__(self):
729         return self.__class__(self.func, *_coconut_reversed(self.iter))
730     def __len__(self):
731         return _coconut.len(self.iter)
732     def __repr__(self):
733         return "starmap(%r, %r)" % (self.func, self.iter)
734     def __reduce__(self):
735         return (self.__class__, (self.func, self.iter))
736     def __reduce_ex__(self, _):
737         return self.__reduce__()
738     def __copy__(self):
739         return self.__class__(self.func, _coconut.copy.copy(self.iter))
740     def __fmap__(self, func):
741         return self.__class__(_coconut_forward_compose(self.func, func), self.iter)
742     def makedata(data_type, *args):
743         """Construct an object of the given data_type containing the given arguments"""
744         if _coconut.hasattr(data_type, "_make") and _coconut.issubclass(data_type, _coconut.object):
745             return data_type._make(args)
746         if _coconut.issubclass(data_type, (_coconut.map, _coconut.range, _coconut.abc.Sequence)):
747             return args
748         if _coconut.issubclass(data_type, _coconut.str):
749             return "".join(args)
750         return data_type(args)
751     def fmap(func, obj):
752         """fmap(func, obj) creates a copy of obj with func applied to its contents.
753         Override by defining obj.__fmap__(func)."""
754         if _coconut.hasattr(obj, "__fmap__"):
755             return obj.__fmap__(func)
756         if obj.__class__.__module__ == "numpy":
757             from numpy import vectorize
758             return vectorize(func)(obj)

```

```

759     return _coconut_makedata(obj.__class__, *(_coconut_starmap(func, obj.items())))
760 def memoize(maxsize=None, *args, **kwargs):
761     """Decorator that memoizes a function,
762     preventing it from being recomputed if it is called multiple times with the
763     return _coconut.functools.lru_cache(maxsize, *args, **kwargs)
764 _coconut_MatchError, _coconut_count, _coconut_enumerate, _coconut_makedata, _coconut
765
766 # Compiled Coconut: -----
767
768 @_coconut_tco
769 @_coconut_mark_as_match
770 def sieve(*_coconut_match_to_args, **_coconut_match_to_kwargs):
771     _coconut_match_check = False
772     _coconut_FunctionMatchError = _coconut_get_function_match_error()
773     if (_coconut.len(_coconut_match_to_args) == 1) and (_coconut.isinstance(_coconut
774         numbers = _coconut.iter(_coconut_match_to_args[0])
775         _coconut_match_temp_0 = _coconut.tuple(_coconut_igetitem(numbers, _coconut
776         if (_coconut.len(_coconut_match_temp_0) == 1) and (not _coconut_match_to_l
777             p = _coconut_match_temp_0[0]
778             _coconut_match_check = True
779     if not _coconut_match_check:
780         _coconut_match_val_repr = _coconut.repr(_coconut_match_to_args)
781         _coconut_match_err = _coconut_FunctionMatchError("pattern-matching failed
782         _coconut_match_err.pattern = 'def sieve( (|p|) :: numbers ) ='
783         _coconut_match_err.value = _coconut_match_to_args
784         raise _coconut_match_err
785
786     return _coconut_tail_call(_coconut.itertools.chain.from_iterable, (_coconut_f
787 primes = (_coconut_forward_compose(count, sieve))(2)
788
789 (_coconut_forward_compose(_coconut.functools.partial(takewhile, lambda p: p < 100)

```

Bytecode

line	byte	opcode	arg	argval
9	0	LOAD_CONST	0	(0)
	2	LOAD_CONST	1	('print_function', 'absolute_import',
	4	IMPORT_NAME	0	(__future__)

	6	IMPORT_FROM	1	(print_function)
	8	STORE_NAME	1	(print_function)
	10	IMPORT_FROM	2	(absolute_import)
	12	STORE_NAME	2	(absolute_import)
	14	IMPORT_FROM	3	(unicode_literals)
	16	STORE_NAME	3	(unicode_literals)
	18	IMPORT_FROM	4	(division)
	20	STORE_NAME	4	(division)
	22	POP_TOP		
10	24	LOAD_CONST	0	(0)
	26	LOAD_CONST	2	(None)
	28	IMPORT_NAME	5	(sys)
	30	STORE_NAME	6	(_coconut_sys)
11	32	LOAD_NAME	6	(_coconut_sys)
	34	LOAD_ATTR	7	(version_info)
	36	LOAD_CONST	3	((3,))
	38	COMPARE_OP	0	(<)
	40	EXTENDED_ARG	1	
	42	POP_JUMP_IF_FALSE	490	
12	44	LOAD_CONST	0	(0)
	46	LOAD_CONST	4	((('chr', 'filter', 'hex', 'input', 'int
	48	IMPORT_NAME	8	(__builtin__)
	50	IMPORT_FROM	9	(chr)
	52	STORE_NAME	9	(chr)
	54	IMPORT_FROM	10	(filter)
	56	STORE_NAME	10	(filter)
	58	IMPORT_FROM	11	(hex)
	60	STORE_NAME	11	(hex)
	62	IMPORT_FROM	12	(input)
	64	STORE_NAME	12	(input)
	66	IMPORT_FROM	13	(int)
	68	STORE_NAME	13	(int)
	70	IMPORT_FROM	14	(map)
	72	STORE_NAME	14	(map)
	74	IMPORT_FROM	15	(object)

	76 STORE_NAME	15 (object)
	78 IMPORT_FROM	16 (oct)
	80 STORE_NAME	16 (oct)
	82 IMPORT_FROM	17 (open)
	84 STORE_NAME	17 (open)
	86 IMPORT_FROM	18 (print)
	88 STORE_NAME	18 (print)
	90 IMPORT_FROM	19 (range)
	92 STORE_NAME	19 (range)
	94 IMPORT_FROM	20 (str)
	96 STORE_NAME	20 (str)
	98 IMPORT_FROM	21 (zip)
	100 STORE_NAME	21 (zip)
	102 IMPORT_FROM	10 (filter)
	104 STORE_NAME	10 (filter)
	106 IMPORT_FROM	22 (reversed)
	108 STORE_NAME	22 (reversed)
	110 IMPORT_FROM	23 (enumerate)
	112 STORE_NAME	23 (enumerate)
	114 IMPORT_FROM	24 (raw_input)
	116 STORE_NAME	24 (raw_input)
	118 IMPORT_FROM	25 (xrange)
	120 STORE_NAME	25 (xrange)
	122 POP_TOP	
13	124 LOAD_NAME	9 (chr)
	126 LOAD_NAME	11 (hex)
	128 LOAD_NAME	12 (input)
	130 LOAD_NAME	13 (int)
	132 LOAD_NAME	14 (map)
	134 LOAD_NAME	15 (object)
	136 LOAD_NAME	16 (oct)
	138 LOAD_NAME	17 (open)
	140 LOAD_NAME	18 (print)
	142 LOAD_NAME	19 (range)
	144 LOAD_NAME	20 (str)
	146 LOAD_NAME	21 (zip)
	148 LOAD_NAME	10 (filter)

	150	LOAD_NAME	22	(reversed)
	152	LOAD_NAME	23	(enumerate)
	154	LOAD_NAME	24	(raw_input)
	156	LOAD_NAME	25	(xrange)
	158	LOAD_NAME	26	(repr)
	160	BUILD_TUPLE	18	
	162	UNPACK_SEQUENCE	18	
	164	STORE_NAME	27	(py_chr)
	166	STORE_NAME	28	(py_hex)
	168	STORE_NAME	29	(py_input)
	170	STORE_NAME	30	(py_int)
	172	STORE_NAME	31	(py_map)
	174	STORE_NAME	32	(py_object)
	176	STORE_NAME	33	(py_oct)
	178	STORE_NAME	34	(py_open)
	180	STORE_NAME	35	(py_print)
	182	STORE_NAME	36	(py_range)
	184	STORE_NAME	37	(py_str)
	186	STORE_NAME	38	(py_zip)
	188	STORE_NAME	39	(py_filter)
	190	STORE_NAME	40	(py_reversed)
	192	STORE_NAME	41	(py_enumerate)
	194	STORE_NAME	42	(py_raw_input)
	196	STORE_NAME	43	(py_xrange)
	198	STORE_NAME	44	(py_repr)
14	200	LOAD_NAME	45	(NotImplemented)
	202	LOAD_NAME	24	(raw_input)
	204	LOAD_NAME	25	(xrange)
	206	LOAD_NAME	13	(int)
	208	LOAD_NAME	46	(long)
	210	LOAD_NAME	18	(print)
	212	LOAD_NAME	20	(str)
	214	LOAD_NAME	47	(unicode)
	216	LOAD_NAME	26	(repr)
	218	BUILD_TUPLE	9	
	220	UNPACK_SEQUENCE	9	
	222	STORE_NAME	48	(_coconut_NotImplemented)

	224 STORE_NAME	49 (_coconut_raw_input)
	226 STORE_NAME	50 (_coconut_xrange)
	228 STORE_NAME	51 (_coconut_int)
	230 STORE_NAME	52 (_coconut_long)
	232 STORE_NAME	53 (_coconut_print)
	234 STORE_NAME	54 (_coconut_str)
	236 STORE_NAME	55 (_coconut_unicode)
	238 STORE_NAME	56 (_coconut_repr)
15	240 LOAD_CONST	0 (0)
	242 LOAD_CONST	5 (('*',))
	244 IMPORT_NAME	57 (future_builtins)
	246 IMPORT_STAR	
16	248 LOAD_NAME	58 (unichr)
	250 LOAD_NAME	47 (unicode)
	252 ROT_TWO	
	254 STORE_NAME	9 (chr)
	256 STORE_NAME	20 (str)
17	258 LOAD_CONST	0 (0)
	260 LOAD_CONST	6 (('open',))
	262 IMPORT_NAME	59 (io)
	264 IMPORT_FROM	17 (open)
	266 STORE_NAME	17 (open)
	268 POP_TOP	
18	270 LOAD_BUILD_CLASS	
	272 LOAD_CONST	7 (<code object object at 0x7f6cf328c190,
	274 LOAD_CONST	8 ('object')
	276 MAKE_FUNCTION	0
	278 LOAD_CONST	8 ('object')
	280 LOAD_NAME	15 (object)
	282 CALL_FUNCTION	3
	284 STORE_NAME	15 (object)
25	286 LOAD_BUILD_CLASS	
	288 LOAD_CONST	9 (<code object int at 0x7f6cf328c500, fi

	290	LOAD_CONST	10 ('int')
	292	MAKE_FUNCTION	0
	294	LOAD_CONST	10 ('int')
	296	LOAD_NAME	51 (_coconut_int)
	298	CALL_FUNCTION	3
	300	STORE_NAME	13 (int)
34	302	LOAD_BUILD_CLASS	
	304	LOAD_CONST	11 (<code object range at 0x7f6cf3299030,
	306	LOAD_CONST	12 ('range')
	308	MAKE_FUNCTION	0
	310	LOAD_CONST	12 ('range')
	312	LOAD_NAME	15 (object)
	314	CALL_FUNCTION	3
	316	STORE_NAME	19 (range)
78	318	LOAD_CONST	0 (0)
	320	LOAD_CONST	13 (('Sequence',))
	322	IMPORT_NAME	60 (collections)
	324	IMPORT_FROM	61 (Sequence)
	326	STORE_NAME	62 (_coconut_Sequence)
	328	POP_TOP	
79	330	LOAD_NAME	62 (_coconut_Sequence)
	332	LOAD_METHOD	63 (register)
	334	LOAD_NAME	19 (range)
	336	CALL_METHOD	1
	338	POP_TOP	
80	340	LOAD_CONST	0 (0)
	342	LOAD_CONST	14 (('wraps',))
	344	IMPORT_NAME	64 (functools)
	346	IMPORT_FROM	65 (wraps)
	348	STORE_NAME	66 (_coconut_wraps)
	350	POP_TOP	
81	352	LOAD_NAME	66 (_coconut_wraps)
	354	LOAD_NAME	53 (_coconut_print)

	356 CALL_FUNCTION	1
82	358 LOAD_CONST	15 (<code object print at 0x7f6cf32993a0,
	360 LOAD_CONST	16 ('print')
	362 MAKE_FUNCTION	0
	364 CALL_FUNCTION	1
	366 STORE_NAME	18 (print)
93	368 LOAD_NAME	66 (_coconut_wraps)
	370 LOAD_NAME	49 (_coconut_raw_input)
	372 CALL_FUNCTION	1
94	374 LOAD_CONST	17 (<code object input at 0x7f6cf32992f0,
	376 LOAD_CONST	18 ('input')
	378 MAKE_FUNCTION	0
	380 CALL_FUNCTION	1
	382 STORE_NAME	12 (input)
98	384 LOAD_NAME	66 (_coconut_wraps)
	386 LOAD_NAME	56 (_coconut_repr)
	388 CALL_FUNCTION	1
99	390 LOAD_CONST	19 (<code object repr at 0x7f6cf3299450, f
	392 LOAD_CONST	20 ('repr')
	394 MAKE_FUNCTION	0
	396 CALL_FUNCTION	1
	398 STORE_NAME	26 (repr)
105	400 LOAD_NAME	26 (repr)
	402 STORE_NAME	67 (ascii)
106	404 LOAD_CONST	21 (<code object raw_input at 0x7f6cf32995
	406 LOAD_CONST	22 ('raw_input')
	408 MAKE_FUNCTION	0
	410 STORE_NAME	24 (raw_input)
109	412 LOAD_CONST	23 (<code object xrange at 0x7f6cf32995b0,
	414 LOAD_CONST	24 ('xrange')

	416	MAKE_FUNCTION	0
	418	STORE_NAME	25 (xrange)
112	420	LOAD_NAME	6 (_coconut_sys)
	422	LOAD_ATTR	7 (version_info)
	424	LOAD_CONST	25 ((2, 7))
	426	COMPARE_OP	0 (<)
	428	EXTENDED_ARG	2
	430	POP_JUMP_IF_FALSE	634
113	432	LOAD_CONST	0 (0)
	434	LOAD_CONST	2 (None)
	436	IMPORT_NAME	64 (functools)
	438	STORE_NAME	68 (_coconut_functools)
	440	LOAD_CONST	0 (0)
	442	LOAD_CONST	2 (None)
	444	IMPORT_NAME	69 (copy_reg)
	446	STORE_NAME	70 (_coconut_copy_reg)
114	448	LOAD_CONST	26 (<code object _coconut_new_partial at 0x...)
	450	LOAD_CONST	27 ('_coconut_new_partial')
	452	MAKE_FUNCTION	0
	454	STORE_NAME	71 (_coconut_new_partial)
116	456	LOAD_NAME	70 (_coconut_copy_reg)
	458	LOAD_METHOD	72 (constructor)
	460	LOAD_NAME	71 (_coconut_new_partial)
	462	CALL_METHOD	1
	464	POP_TOP	
117	466	LOAD_CONST	28 (<code object _coconut_reduce_partial at 0x...)
	468	LOAD_CONST	29 ('_coconut_reduce_partial')
	470	MAKE_FUNCTION	0
	472	STORE_NAME	73 (_coconut_reduce_partial)
119	474	LOAD_NAME	70 (_coconut_copy_reg)
	476	LOAD_METHOD	74 (pickle)
	478	LOAD_NAME	68 (_coconut_functools)

	480	LOAD_ATTR	75	(partial)
	482	LOAD_NAME	73	(_coconut_reduce_partial)
	484	CALL_METHOD	2	
	486	POP_TOP		
	488	JUMP_FORWARD	144	(to 634)
121	>>	490	LOAD_CONST	0 (0)
		492	LOAD_CONST	30 (('chr', 'filter', 'hex', 'input', 'int
		494	IMPORT_NAME	76 (builtins)
		496	IMPORT_FROM	9 (chr)
		498	STORE_NAME	9 (chr)
		500	IMPORT_FROM	10 (filter)
		502	STORE_NAME	10 (filter)
		504	IMPORT_FROM	11 (hex)
		506	STORE_NAME	11 (hex)
		508	IMPORT_FROM	12 (input)
		510	STORE_NAME	12 (input)
		512	IMPORT_FROM	13 (int)
		514	STORE_NAME	13 (int)
		516	IMPORT_FROM	14 (map)
		518	STORE_NAME	14 (map)
		520	IMPORT_FROM	15 (object)
		522	STORE_NAME	15 (object)
		524	IMPORT_FROM	16 (oct)
		526	STORE_NAME	16 (oct)
		528	IMPORT_FROM	17 (open)
		530	STORE_NAME	17 (open)
		532	IMPORT_FROM	18 (print)
		534	STORE_NAME	18 (print)
		536	IMPORT_FROM	19 (range)
		538	STORE_NAME	19 (range)
		540	IMPORT_FROM	20 (str)
		542	STORE_NAME	20 (str)
		544	IMPORT_FROM	21 (zip)
		546	STORE_NAME	21 (zip)
		548	IMPORT_FROM	10 (filter)
		550	STORE_NAME	10 (filter)
		552	IMPORT_FROM	22 (reversed)

	554	STORE_NAME	22	(reversed)
	556	IMPORT_FROM	23	(enumerate)
	558	STORE_NAME	23	(enumerate)
	560	POP_TOP		
122	562	LOAD_NAME	9	(chr)
	564	LOAD_NAME	11	(hex)
	566	LOAD_NAME	12	(input)
	568	LOAD_NAME	13	(int)
	570	LOAD_NAME	14	(map)
	572	LOAD_NAME	15	(object)
	574	LOAD_NAME	16	(oct)
	576	LOAD_NAME	17	(open)
	578	LOAD_NAME	18	(print)
	580	LOAD_NAME	19	(range)
	582	LOAD_NAME	20	(str)
	584	LOAD_NAME	21	(zip)
	586	LOAD_NAME	10	(filter)
	588	LOAD_NAME	22	(reversed)
	590	LOAD_NAME	23	(enumerate)
	592	LOAD_NAME	26	(repr)
	594	BUILD_TUPLE	16	
	596	UNPACK_SEQUENCE	16	
	598	STORE_NAME	27	(py_chr)
	600	STORE_NAME	28	(py_hex)
	602	STORE_NAME	29	(py_input)
	604	STORE_NAME	30	(py_int)
	606	STORE_NAME	31	(py_map)
	608	STORE_NAME	32	(py_object)
	610	STORE_NAME	33	(py_oct)
	612	STORE_NAME	34	(py_open)
	614	STORE_NAME	35	(py_print)
	616	STORE_NAME	36	(py_range)
	618	STORE_NAME	37	(py_str)
	620	STORE_NAME	38	(py_zip)
	622	STORE_NAME	39	(py_filter)
	624	STORE_NAME	40	(py_reversed)
	626	STORE_NAME	41	(py_enumerate)

	628	STORE_NAME	44	(py_repr)
123	630	LOAD_NAME	20	(str)
	632	STORE_NAME	54	(_coconut_str)
124	>>	634 LOAD_BUILD_CLASS		
	636	LOAD_CONST	31	(<code object _coconut at 0x7f6cf3299b3
	638	LOAD_CONST	32	('_coconut')
	640	MAKE_FUNCTION	0	
	642	LOAD_CONST	32	('_coconut')
	644	LOAD_NAME	15	(object)
	646	CALL_FUNCTION	3	
	648	STORE_NAME	77	(_coconut)
148	650	LOAD_NAME	77	(_coconut)
	652	LOAD_METHOD	15	(object)
	654	CALL_METHOD	0	
	656	STORE_NAME	78	(_coconut_sentinel)
149	658	LOAD_BUILD_CLASS		
	660	LOAD_CONST	33	(<code object MatchError at 0x7f6cf3299
	662	LOAD_CONST	34	('MatchError')
	664	MAKE_FUNCTION	0	
	666	LOAD_CONST	34	('MatchError')
	668	LOAD_NAME	79	(Exception)
	670	CALL_FUNCTION	3	
	672	STORE_NAME	80	(MatchError)
152	674	LOAD_BUILD_CLASS		
	676	LOAD_CONST	35	(<code object _coconut_tail_call at 0x7
	678	LOAD_CONST	36	('_coconut_tail_call')
	680	MAKE_FUNCTION	0	
	682	LOAD_CONST	36	('_coconut_tail_call')
	684	LOAD_NAME	15	(object)
	686	CALL_FUNCTION	3	
	688	STORE_NAME	81	(_coconut_tail_call)
156	690	BUILD_MAP	0	

	692 STORE_NAME	82 (_coconut_tco_func_dict)
157	694 LOAD_CONST	37 (<code object _coconut_tco at 0x7f6cf32
	696 LOAD_CONST	38 ('_coconut_tco')
	698 MAKE_FUNCTION	0
	700 STORE_NAME	83 (_coconut_tco)
175	702 LOAD_CONST	39 (<code object _coconut_igetitem at 0x7f
	704 LOAD_CONST	40 ('_coconut_igetitem')
	706 MAKE_FUNCTION	0
	708 STORE_NAME	84 (_coconut_igetitem)
190	710 LOAD_BUILD_CLASS	
	712 LOAD_CONST	41 (<code object _coconut_base_compose at
	714 LOAD_CONST	42 ('_coconut_base_compose')
	716 MAKE_FUNCTION	0
	718 LOAD_CONST	42 ('_coconut_base_compose')
	720 LOAD_NAME	15 (object)
	722 CALL_FUNCTION	3
	724 STORE_NAME	85 (_coconut_base_compose)
219	726 LOAD_CONST	43 (<code object _coconut_forward_compose
	728 LOAD_CONST	44 ('_coconut_forward_compose')
	730 MAKE_FUNCTION	0
	732 STORE_NAME	86 (_coconut_forward_compose)
220	734 LOAD_CONST	45 (<code object _coconut_back_compose at
	736 LOAD_CONST	46 ('_coconut_back_compose')
	738 MAKE_FUNCTION	0
	740 STORE_NAME	87 (_coconut_back_compose)
221	742 LOAD_CONST	47 (<code object _coconut_forward_star_com
	744 LOAD_CONST	48 ('_coconut_forward_star_compose')
	746 MAKE_FUNCTION	0
	748 STORE_NAME	88 (_coconut_forward_star_compose)
222	750 LOAD_CONST	49 (<code object _coconut_back_star_compos
	752 LOAD_CONST	50 ('_coconut_back_star_compose')

	754 MAKE_FUNCTION	0
	756 STORE_NAME	89 (_coconut_back_star_compose)
223	758 LOAD_CONST	51 (<code object _coconut_forward_dubstar_compose at 0x7f6cf3b1b1b1>)
	760 LOAD_CONST	52 ('_coconut_forward_dubstar_compose')
	762 MAKE_FUNCTION	0
	764 STORE_NAME	90 (_coconut_forward_dubstar_compose)
224	766 LOAD_CONST	53 (<code object _coconut_back_dubstar_compose at 0x7f6cf3b1b1b1>)
	768 LOAD_CONST	54 ('_coconut_back_dubstar_compose')
	770 MAKE_FUNCTION	0
	772 STORE_NAME	91 (_coconut_back_dubstar_compose)
225	774 LOAD_CONST	55 (<code object _coconut_pipe at 0x7f6cf3b1b1b1>)
	776 LOAD_CONST	56 ('_coconut_pipe')
	778 MAKE_FUNCTION	0
	780 STORE_NAME	92 (_coconut_pipe)
226	782 LOAD_CONST	57 (<code object _coconut_star_pipe at 0x7f6cf3b1b1b1>)
	784 LOAD_CONST	58 ('_coconut_star_pipe')
	786 MAKE_FUNCTION	0
	788 STORE_NAME	93 (_coconut_star_pipe)
227	790 LOAD_CONST	59 (<code object _coconut_dubstar_pipe at 0x7f6cf3b1b1b1>)
	792 LOAD_CONST	60 ('_coconut_dubstar_pipe')
	794 MAKE_FUNCTION	0
	796 STORE_NAME	94 (_coconut_dubstar_pipe)
228	798 LOAD_CONST	61 (<code object _coconut_back_pipe at 0x7f6cf3b1b1b1>)
	800 LOAD_CONST	62 ('_coconut_back_pipe')
	802 MAKE_FUNCTION	0
	804 STORE_NAME	95 (_coconut_back_pipe)
229	806 LOAD_CONST	63 (<code object _coconut_back_star_pipe at 0x7f6cf3b1b1b1>)
	808 LOAD_CONST	64 ('_coconut_back_star_pipe')
	810 MAKE_FUNCTION	0
	812 STORE_NAME	96 (_coconut_back_star_pipe)

230	814 LOAD_CONST	65 (<code object _coconut_back_dubstar_pipe
	816 LOAD_CONST	66 ('_coconut_back_dubstar_pipe')
	818 MAKE_FUNCTION	0
	820 STORE_NAME	97 (_coconut_back_dubstar_pipe)
231	822 LOAD_CONST	129 ((None,))
	824 LOAD_CONST	67 (<code object _coconut_assert at 0x7f6c
	826 LOAD_CONST	68 ('_coconut_assert')
	828 MAKE_FUNCTION	1 (defaults)
	830 STORE_NAME	98 (_coconut_assert)
232	832 LOAD_CONST	69 (<code object _coconut_bool_and at 0x7f
	834 LOAD_CONST	70 ('_coconut_bool_and')
	836 MAKE_FUNCTION	0
	838 STORE_NAME	99 (_coconut_bool_and)
233	840 LOAD_CONST	71 (<code object _coconut_bool_or at 0x7f6
	842 LOAD_CONST	72 ('_coconut_bool_or')
	844 MAKE_FUNCTION	0
	846 STORE_NAME	100 (_coconut_bool_or)
234	848 LOAD_CONST	73 (<code object _coconut_none_coalesce at
	850 LOAD_CONST	74 ('_coconut_none_coalesce')
	852 MAKE_FUNCTION	0
	854 STORE_NAME	101 (_coconut_none_coalesce)
235	856 LOAD_CONST	75 (<code object _coconut_minus at 0x7f6cf
	858 LOAD_CONST	76 ('_coconut_minus')
	860 MAKE_FUNCTION	0
	862 STORE_NAME	102 (_coconut_minus)
241	864 LOAD_NAME	77 (_coconut)
	866 LOAD_ATTR	64 (functools)
	868 LOAD_METHOD	65 (wraps)
	870 LOAD_NAME	77 (_coconut)
	872 LOAD_ATTR	103 (itertools)
	874 LOAD_ATTR	104 (tee)
	876 CALL_METHOD	1

242	878 LOAD_CONST	130 ((2,))
	880 LOAD_CONST	78 (<code object tee at 0x7f6cf329e5b0, fi
	882 LOAD_CONST	79 ('tee')
	884 MAKE_FUNCTION	1 (defaults)
	886 CALL_FUNCTION	1
	888 STORE_NAME	104 (tee)
248	890 LOAD_BUILD_CLASS	
	892 LOAD_CONST	80 (<code object reiterable at 0x7f6cf329e
	894 LOAD_CONST	81 ('reiterable')
	896 MAKE_FUNCTION	0
	898 LOAD_CONST	81 ('reiterable')
	900 LOAD_NAME	15 (object)
	902 CALL_FUNCTION	3
	904 STORE_NAME	105 (reiterable)
272	906 LOAD_BUILD_CLASS	
	908 LOAD_CONST	82 (<code object scan at 0x7f6cf32a3240, f
	910 LOAD_CONST	83 ('scan')
	912 MAKE_FUNCTION	0
	914 LOAD_CONST	83 ('scan')
	916 LOAD_NAME	15 (object)
	918 CALL_FUNCTION	3
	920 STORE_NAME	106 (scan)
300	922 LOAD_BUILD_CLASS	
	924 LOAD_CONST	84 (<code object reversed at 0x7f6cf32a3d4
	926 LOAD_CONST	85 ('reversed')
	928 MAKE_FUNCTION	0
	930 LOAD_CONST	85 ('reversed')
	932 LOAD_NAME	15 (object)
	934 CALL_FUNCTION	3
	936 STORE_NAME	22 (reversed)
342	938 LOAD_BUILD_CLASS	
	940 LOAD_CONST	86 (<code object map at 0x7f6cf32a6920, fi
	942 LOAD_CONST	87 ('map')

	944	MAKE_FUNCTION	0
	946	LOAD_CONST	87 ('map')
	948	LOAD_NAME	77 (_coconut)
	950	LOAD_ATTR	14 (map)
	952	CALL_FUNCTION	3
	954	STORE_NAME	14 (map)
369	956	LOAD_BUILD_CLASS	
	958	LOAD_CONST	88 (<code object parallel_map at 0x7f6cf32ab3a0, fi
	960	LOAD_CONST	89 ('parallel_map')
	962	MAKE_FUNCTION	0
	964	LOAD_CONST	89 ('parallel_map')
	966	LOAD_NAME	14 (map)
	968	CALL_FUNCTION	3
	970	STORE_NAME	107 (parallel_map)
379	972	LOAD_BUILD_CLASS	
	974	LOAD_CONST	90 (<code object concurrent_map at 0x7f6cf32ab3a0, fi
	976	LOAD_CONST	91 ('concurrent_map')
	978	MAKE_FUNCTION	0
	980	LOAD_CONST	91 ('concurrent_map')
	982	LOAD_NAME	14 (map)
	984	CALL_FUNCTION	3
	986	STORE_NAME	108 (concurrent_map)
389	988	LOAD_BUILD_CLASS	
	990	LOAD_CONST	92 (<code object filter at 0x7f6cf32ab3a0, fi
	992	LOAD_CONST	93 ('filter')
	994	MAKE_FUNCTION	0
	996	LOAD_CONST	93 ('filter')
	998	LOAD_NAME	77 (_coconut)
	1000	LOAD_ATTR	10 (filter)
	1002	CALL_FUNCTION	3
	1004	STORE_NAME	10 (filter)
410	1006	LOAD_BUILD_CLASS	
	1008	LOAD_CONST	94 (<code object zip at 0x7f6cf32ab3a0, fi
	1010	LOAD_CONST	95 ('zip')

	1012	MAKE_FUNCTION	0
	1014	LOAD_CONST	95 ('zip')
	1016	LOAD_NAME	77 (_coconut)
	1018	LOAD_ATTR	21 (zip)
	1020	CALL_FUNCTION	3
	1022	STORE_NAME	21 (zip)
436	1024	LOAD_BUILD_CLASS	
	1026	LOAD_CONST	96 (<code object enumerate at 0x7f6cf32ac6)
	1028	LOAD_CONST	97 ('enumerate')
	1030	MAKE_FUNCTION	0
	1032	LOAD_CONST	97 ('enumerate')
	1034	LOAD_NAME	77 (_coconut)
	1036	LOAD_ATTR	23 (enumerate)
	1038	CALL_FUNCTION	3
	1040	STORE_NAME	23 (enumerate)
461	1042	LOAD_BUILD_CLASS	
	1044	LOAD_CONST	98 (<code object count at 0x7f6cf32b1190,
	1046	LOAD_CONST	99 ('count')
	1048	MAKE_FUNCTION	0
	1050	LOAD_CONST	99 ('count')
	1052	LOAD_NAME	15 (object)
	1054	CALL_FUNCTION	3
	1056	STORE_NAME	109 (count)
520	1058	LOAD_BUILD_CLASS	
	1060	LOAD_CONST	100 (<code object groupsof at 0x7f6cf32b166)
	1062	LOAD_CONST	101 ('groupsof')
	1064	MAKE_FUNCTION	0
	1066	LOAD_CONST	101 ('groupsof')
	1068	LOAD_NAME	15 (object)
	1070	CALL_FUNCTION	3
	1072	STORE_NAME	110 (groupsof)
555	1074	LOAD_BUILD_CLASS	
	1076	LOAD_CONST	102 (<code object recursive_iterator at 0x7f6cf32b166)
	1078	LOAD_CONST	103 ('recursive_iterator')

	1080 MAKE_FUNCTION	0
	1082 LOAD_CONST	103 ('recursive_iterator')
	1084 LOAD_NAME	15 (object)
	1086 CALL_FUNCTION	3
	1088 STORE_NAME	111 (recursive_iterator)
594	1090 LOAD_BUILD_CLASS	
	1092 LOAD_CONST	104 (<code object _coconut_FunctionMatchErrorContext at 0x7f6cf32c1000>)
	1094 LOAD_CONST	105 ('_coconut_FunctionMatchErrorContext')
	1096 MAKE_FUNCTION	0
	1098 LOAD_CONST	105 ('_coconut_FunctionMatchErrorContext')
	1100 LOAD_NAME	15 (object)
	1102 CALL_FUNCTION	3
	1104 STORE_NAME	112 (_coconut_FunctionMatchErrorContext)
617	1106 LOAD_NAME	112 (_coconut_FunctionMatchErrorContext)
	1108 LOAD_ATTR	113 (get)
	1110 STORE_NAME	114 (_coconut_get_function_match_error)
618	1112 LOAD_BUILD_CLASS	
	1114 LOAD_CONST	106 (<code object _coconut_base_pattern_func at 0x7f6cf32c1000>)
	1116 LOAD_CONST	107 ('_coconut_base_pattern_func')
	1118 MAKE_FUNCTION	0
	1120 LOAD_CONST	107 ('_coconut_base_pattern_func')
	1122 LOAD_NAME	15 (object)
	1124 CALL_FUNCTION	3
	1126 STORE_NAME	115 (_coconut_base_pattern_func)
657	1128 LOAD_CONST	108 (<code object _coconut_mark_as_match at 0x7f6cf32c1000>)
	1130 LOAD_CONST	109 ('_coconut_mark_as_match')
	1132 MAKE_FUNCTION	0
	1134 STORE_NAME	116 (_coconut_mark_as_match)
660	1136 LOAD_CONST	110 (<code object addpattern at 0x7f6cf32c1000>)
	1138 LOAD_CONST	111 ('addpattern')
	1140 MAKE_FUNCTION	0
	1142 STORE_NAME	117 (addpattern)

669	1144 LOAD_NAME	117 (addpattern)
	1146 STORE_NAME	118 (_coconut_addpattern)
670	1148 LOAD_BUILD_CLASS	
	1150 LOAD_CONST	112 (<code object _coconut_partial at 0x7f6cf32c1df0)
	1152 LOAD_CONST	113 ('_coconut_partial')
	1154 MAKE_FUNCTION	0
	1156 LOAD_CONST	113 ('_coconut_partial')
	1158 LOAD_NAME	15 (object)
	1160 CALL_FUNCTION	3
	1162 STORE_NAME	119 (_coconut_partial)
712	1164 LOAD_CONST	131 ((0,))
	1166 LOAD_CONST	114 (<code object consume at 0x7f6cf32c1df0)
	1168 LOAD_CONST	115 ('consume')
	1170 MAKE_FUNCTION	1 (defaults)
	1172 STORE_NAME	120 (consume)
715	1174 LOAD_BUILD_CLASS	
	1176 LOAD_CONST	116 (<code object starmap at 0x7f6cf32c25b0)
	1178 LOAD_CONST	117 ('starmap')
	1180 MAKE_FUNCTION	0
	1182 LOAD_CONST	117 ('starmap')
	1184 LOAD_NAME	77 (_coconut)
	1186 LOAD_ATTR	103 (itertools)
	1188 LOAD_ATTR	121 (starmap)
	1190 CALL_FUNCTION	3
	1192 STORE_NAME	121 (starmap)
742	1194 LOAD_CONST	118 (<code object makedata at 0x7f6cf32c25b0)
	1196 LOAD_CONST	119 ('makedata')
	1198 MAKE_FUNCTION	0
	1200 STORE_NAME	122 (makedata)
751	1202 LOAD_CONST	120 (<code object fmap at 0x7f6cf32c2710, f...)
	1204 LOAD_CONST	121 ('fmap')
	1206 MAKE_FUNCTION	0
	1208 STORE_NAME	123 (fmap)

760	1210 LOAD_CONST	132 ((None,))
	1212 LOAD_CONST	122 (<code object memoize at 0x7f6cf32c2660)
	1214 LOAD_CONST	123 ('memoize')
	1216 MAKE_FUNCTION	1 (defaults)
	1218 STORE_NAME	124 (memoize)
764	1220 LOAD_NAME	80 (MatchError)
	1222 LOAD_NAME	109 (count)
	1224 LOAD_NAME	23 (enumerate)
	1226 LOAD_NAME	122 (makedata)
	1228 LOAD_NAME	14 (map)
	1230 LOAD_NAME	22 (reversed)
	1232 LOAD_NAME	121 (starmap)
	1234 LOAD_NAME	104 (tee)
	1236 LOAD_NAME	21 (zip)
	1238 LOAD_CONST	124 (False)
	1240 LOAD_NAME	77 (_coconut)
	1242 LOAD_ATTR	64 (functools)
	1244 LOAD_ATTR	125 (reduce)
	1246 LOAD_NAME	77 (_coconut)
	1248 LOAD_ATTR	103 (itertools)
	1250 LOAD_ATTR	126 (takewhile)
	1252 LOAD_NAME	77 (_coconut)
	1254 LOAD_ATTR	103 (itertools)
	1256 LOAD_ATTR	127 (dropwhile)
	1258 BUILD_TUPLE	13
	1260 UNPACK_SEQUENCE	13
	1262 STORE_NAME	128 (_coconut_MatchError)
	1264 STORE_NAME	129 (_coconut_count)
	1266 STORE_NAME	130 (_coconut_enumerate)
	1268 STORE_NAME	131 (_coconut_makedata)
	1270 STORE_NAME	132 (_coconut_map)
	1272 STORE_NAME	133 (_coconut_reversed)
	1274 STORE_NAME	134 (_coconut_starmap)
	1276 STORE_NAME	135 (_coconut_tee)
	1278 STORE_NAME	136 (_coconut_zip)
	1280 STORE_NAME	137 (TYPE_CHECKING)

	1282 STORE_NAME	125 (reduce)
	1284 STORE_NAME	126 (takewhile)
	1286 STORE_NAME	127 (dropwhile)
768	1288 LOAD_NAME	83 (_coconut_tco)
769	1290 LOAD_NAME	116 (_coconut_mark_as_match)
770	1292 LOAD_CONST	125 (<code object sieve at 0x7f6cf32c2c90,
	1294 LOAD_CONST	126 ('sieve')
	1296 MAKE_FUNCTION	0
	1298 CALL_FUNCTION	1
	1300 CALL_FUNCTION	1
	1302 STORE_NAME	138 (sieve)
787	1304 LOAD_NAME	86 (_coconut_forward_compose)
	1306 LOAD_NAME	109 (count)
	1308 LOAD_NAME	138 (sieve)
	1310 CALL_FUNCTION	2
	1312 LOAD_CONST	77 (2)
	1314 CALL_FUNCTION	1
	1316 STORE_NAME	139 (primes)
789	1318 LOAD_NAME	86 (_coconut_forward_compose)
	1320 LOAD_NAME	77 (_coconut)
	1322 LOAD_ATTR	64 (functools)
	1324 LOAD_METHOD	75 (partial)
	1326 LOAD_NAME	126 (takewhile)
	1328 LOAD_CONST	127 (<code object <lambda> at 0x7f6cf32c2d4
	1330 LOAD_CONST	128 ('<lambda>')
	1332 MAKE_FUNCTION	0
	1334 CALL_METHOD	2
	1336 LOAD_NAME	140 (list)
	1338 LOAD_NAME	18 (print)
	1340 CALL_FUNCTION	3
	1342 LOAD_NAME	139 (primes)
	1344 CALL_FUNCTION	1
	1346 POP_TOP	

1348	LOAD_CONST	2 (None)
1350	RETURN_VALUE	

Disassembly of <code object object at 0x7f6cf328c190, file "<dis>", line 18>:

18	0	LOAD_NAME	0 (__name__)
	2	STORE_NAME	1 (__module__)
	4	LOAD_CONST	0 ('object')
	6	STORE_NAME	2 (__qualname__)
19	8	LOAD_CONST	1 (())
	10	STORE_NAME	3 (__slots__)
20	12	LOAD_CONST	2 (<code object __ne__ at 0x7f6cf328c0e0,
	14	LOAD_CONST	3 ('object.__ne__')
	16	MAKE_FUNCTION	0
	18	STORE_NAME	4 (__ne__)
	20	LOAD_CONST	4 (None)
	22	RETURN_VALUE	

Disassembly of <code object __ne__ at 0x7f6cf328c0e0, file "<dis>", line 20>:

21	0	LOAD_FAST	0 (self)
	2	LOAD_FAST	1 (other)
	4	COMPARE_OP	2 (==)
	6	STORE_FAST	2 (eq)
22	8	LOAD_FAST	2 (eq)
	10	LOAD_GLOBAL	0 (_coconut_NotImplemented)
	12	COMPARE_OP	8 (is)
	14	POP_JUMP_IF_FALSE	20
23	16	LOAD_FAST	2 (eq)
	18	RETURN_VALUE	
24	>>	20	LOAD_FAST
		22	UNARY_NOT
		24	RETURN_VALUE

Disassembly of <code object int at 0x7f6cf328c500, file "<dis>", line 25>:

25	0 LOAD_NAME	0 (__name__)
	2 STORE_NAME	1 (__module__)
	4 LOAD_CONST	0 ('int')
	6 STORE_NAME	2 (__qualname__)
26	8 LOAD_CONST	1 (())
	10 STORE_NAME	3 (__slots__)
27	12 LOAD_NAME	4 (hasattr)
	14 LOAD_NAME	5 (_coconut_int)
	16 LOAD_CONST	2 ('__doc__')
	18 CALL_FUNCTION	2
	20 POP_JUMP_IF_FALSE	28
28	22 LOAD_NAME	5 (_coconut_int)
	24 LOAD_ATTR	6 (__doc__)
	26 STORE_NAME	6 (__doc__)
29	>> 28 LOAD_BUILD_CLASS	
	30 LOAD_CONST	3 (<code object __metaclass__ at 0x7f6cf328c450, file "<dis>", line 29)
	32 LOAD_CONST	4 ('__metaclass__')
	34 MAKE_FUNCTION	0
	36 LOAD_CONST	4 ('__metaclass__')
	38 LOAD_NAME	7 (type)
	40 CALL_FUNCTION	3
	42 STORE_NAME	8 (__metaclass__)
	44 LOAD_CONST	5 (None)
	46 RETURN_VALUE	

Disassembly of <code object __metaclass__ at 0x7f6cf328c450, file "<dis>", line 29

29	0 LOAD_NAME	0 (__name__)
	2 STORE_NAME	1 (__module__)
	4 LOAD_CONST	0 ('int.__metaclass__')
	6 STORE_NAME	2 (__qualname__)
30	8 LOAD_CONST	1 (<code object __instancecheck__ at 0x7f6cf328c450, file "<dis>", line 30)
	10 LOAD_CONST	2 ('int.__metaclass__.__instancecheck__')
	12 MAKE_FUNCTION	0

	14	STORE_NAME	3	(__instancecheck__)
32	16	LOAD_CONST	3	(<code object __subclasscheck__ at 0x7f6cf328c2f0, file "<dis>", line 33>)
	18	LOAD_CONST	4	('int.__metaclass__.__subclasscheck__')
	20	MAKE_FUNCTION	0	
	22	STORE_NAME	4	(__subclasscheck__)
	24	LOAD_CONST	5	(None)
	26	RETURN_VALUE		

Disassembly of <code object __instancecheck__ at 0x7f6cf328c2f0, file "<dis>", line 31>:

31	0	LOAD_GLOBAL	0	(_coconut)
	2	LOAD_METHOD	1	(isinstance)
	4	LOAD_FAST	1	(inst)
	6	LOAD_GLOBAL	2	(_coconut_int)
	8	LOAD_GLOBAL	3	(_coconut_long)
	10	BUILD_TUPLE	2	
	12	CALL_METHOD	2	
	14	RETURN_VALUE		

Disassembly of <code object __subclasscheck__ at 0x7f6cf328c3a0, file "<dis>", line 33>:

33	0	LOAD_GLOBAL	0	(_coconut)
	2	LOAD_METHOD	1	(issubclass)
	4	LOAD_FAST	1	(subcls)
	6	LOAD_GLOBAL	2	(_coconut_int)
	8	LOAD_GLOBAL	3	(_coconut_long)
	10	BUILD_TUPLE	2	
	12	CALL_METHOD	2	
	14	RETURN_VALUE		

Disassembly of <code object range at 0x7f6cf3299030, file "<dis>", line 34>:

34	0	LOAD_NAME	0	(__name__)
	2	STORE_NAME	1	(__module__)
	4	LOAD_CONST	0	('range')
	6	STORE_NAME	2	(__qualname__)
35	8	LOAD_CONST	1	((('_xrange',),))
	10	STORE_NAME	3	(__slots__)

36	12 LOAD_NAME	4 (hasattr)
	14 LOAD_NAME	5 (_coconut_xrange)
	16 LOAD_CONST	2 ('__doc__')
	18 CALL_FUNCTION	2
	20 POP_JUMP_IF_FALSE	28
37	22 LOAD_NAME	5 (_coconut_xrange)
	24 LOAD_ATTR	6 (__doc__)
	26 STORE_NAME	6 (__doc__)
38	>> 28 LOAD_CONST	3 (<code object __init__ at 0x7f6cf328c5b...)
	30 LOAD_CONST	4 ('range.__init__')
	32 MAKE_FUNCTION	0
	34 STORE_NAME	7 (__init__)
40	36 LOAD_CONST	5 (<code object __iter__ at 0x7f6cf328c6...)
	38 LOAD_CONST	6 ('range.__iter__')
	40 MAKE_FUNCTION	0
	42 STORE_NAME	8 (__iter__)
42	44 LOAD_CONST	7 (<code object __reversed__ at 0x7f6cf32...)
	46 LOAD_CONST	8 ('range.__reversed__')
	48 MAKE_FUNCTION	0
	50 STORE_NAME	9 (__reversed__)
44	52 LOAD_CONST	9 (<code object __len__ at 0x7f6cf328c7c...)
	54 LOAD_CONST	10 ('range.__len__')
	56 MAKE_FUNCTION	0
	58 STORE_NAME	10 (__len__)
46	60 LOAD_CONST	11 (<code object __contains__ at 0x7f6cf32...)
	62 LOAD_CONST	12 ('range.__contains__')
	64 MAKE_FUNCTION	0
	66 STORE_NAME	11 (__contains__)
48	68 LOAD_CONST	13 (<code object __getitem__ at 0x7f6cf32...)
	70 LOAD_CONST	14 ('range.__getitem__')
	72 MAKE_FUNCTION	0

	74 STORE_NAME	12 (__getitem__)
55	76 LOAD_CONST	15 (<code object count at 0x7f6cf328c9d0,
	78 LOAD_CONST	16 ('range.count')
	80 MAKE_FUNCTION	0
	82 STORE_NAME	13 (count)
58	84 LOAD_CONST	17 (<code object index at 0x7f6cf328ca80,
	86 LOAD_CONST	18 ('range.index')
	88 MAKE_FUNCTION	0
	90 STORE_NAME	14 (index)
63	92 LOAD_CONST	19 (<code object __repr__ at 0x7f6cf328cb3
	94 LOAD_CONST	20 ('range.__repr__')
	96 MAKE_FUNCTION	0
	98 STORE_NAME	15 (__repr__)
65	100 LOAD_NAME	16 (property)
66	102 LOAD_CONST	21 (<code object _args at 0x7f6cf328cbe0,
	104 LOAD_CONST	22 ('range._args')
	106 MAKE_FUNCTION	0
	108 CALL_FUNCTION	1
	110 STORE_NAME	17 (_args)
68	112 LOAD_CONST	23 (<code object __reduce_ex__ at 0x7f6cf3
	114 LOAD_CONST	24 ('range.__reduce_ex__')
	116 MAKE_FUNCTION	0
	118 STORE_NAME	18 (__reduce_ex__)
70	120 LOAD_CONST	25 (<code object __reduce__ at 0x7f6cf328c
	122 LOAD_CONST	26 ('range.__reduce__')
	124 MAKE_FUNCTION	0
	126 STORE_NAME	19 (__reduce__)
72	128 LOAD_CONST	27 (<code object __hash__ at 0x7f6cf328cd1
	130 LOAD_CONST	28 ('range.__hash__')
	132 MAKE_FUNCTION	0

	134	STORE_NAME	20	(__hash__)
74	136	LOAD_CONST	29	(<code object __copy__ at 0x7f6cf328cea
	138	LOAD_CONST	30	('range.__copy__')
	140	MAKE_FUNCTION	0	
	142	STORE_NAME	21	(__copy__)
76	144	LOAD_CONST	31	(<code object __eq__ at 0x7f6cf328cf50,
	146	LOAD_CONST	32	('range.__eq__')
	148	MAKE_FUNCTION	0	
	150	STORE_NAME	22	(__eq__)
	152	LOAD_CONST	33	(None)
	154	RETURN_VALUE		

Disassembly of <code object __init__ at 0x7f6cf328c5b0, file "<dis>", line 38>:

39	0	LOAD_GLOBAL	0	(_coconut_xrange)
	2	LOAD_FAST	1	(args)
	4	CALL_FUNCTION_EX	0	
	6	LOAD_FAST	0	(self)
	8	STORE_ATTR	1	(_xrange)
	10	LOAD_CONST	0	(None)
	12	RETURN_VALUE		

Disassembly of <code object __iter__ at 0x7f6cf328c660, file "<dis>", line 40>:

41	0	LOAD_GLOBAL	0	(_coconut)
	2	LOAD_METHOD	1	(iter)
	4	LOAD_FAST	0	(self)
	6	LOAD_ATTR	2	(_xrange)
	8	CALL_METHOD	1	
	10	RETURN_VALUE		

Disassembly of <code object __reversed__ at 0x7f6cf328c710, file "<dis>", line 42>:

43	0	LOAD_GLOBAL	0	(_coconut)
	2	LOAD_METHOD	1	(reversed)
	4	LOAD_FAST	0	(self)
	6	LOAD_ATTR	2	(_xrange)
	8	CALL_METHOD	1	
	10	RETURN_VALUE		

Disassembly of <code object __len__ at 0x7f6cf328c7c0, file "<dis>", line 44>:

45	0	LOAD_GLOBAL	0	(_coconut)
	2	LOAD_METHOD	1	(len)
	4	LOAD_FAST	0	(self)
	6	LOAD_ATTR	2	(_xrange)
	8	CALL_METHOD	1	
	10	RETURN_VALUE		

Disassembly of <code object __contains__ at 0x7f6cf328c870, file "<dis>", line 46>:

47	0	LOAD_FAST	1	(elem)
	2	LOAD_FAST	0	(self)
	4	LOAD_ATTR	0	(_xrange)
	6	COMPARE_OP	6	(in)
	8	RETURN_VALUE		

Disassembly of <code object __getitem__ at 0x7f6cf328c920, file "<dis>", line 48>:

49	0	LOAD_GLOBAL	0	(_coconut)
	2	LOAD_METHOD	1	(isinstance)
	4	LOAD_FAST	1	(index)
	6	LOAD_GLOBAL	0	(_coconut)
	8	LOAD_ATTR	2	(slice)
	10	CALL_METHOD	2	
	12	EXTENDED_ARG	1	
	14	POP_JUMP_IF_FALSE	260	

50	16	LOAD_GLOBAL	0	(_coconut)
	18	LOAD_ATTR	2	(slice)
	20	LOAD_FAST	0	(self)
	22	LOAD_ATTR	3	(_args)
	24	CALL_FUNCTION_EX	0	
	26	STORE_FAST	2	(args)

51	28	LOAD_FAST	2	(args)
	30	LOAD_ATTR	4	(start)
	32	LOAD_CONST	0	(None)
	34	COMPARE_OP	9	(is not)
	36	POP_JUMP_IF_FALSE	44	

	38	LOAD_FAST	2	(args)
	40	LOAD_ATTR	4	(start)
	42	JUMP_FORWARD	2	(to 46)
>>	44	LOAD_CONST	1	(0)
>>	46	LOAD_FAST	2	(args)
	48	LOAD_ATTR	5	(stop)
	50	LOAD_FAST	2	(args)
	52	LOAD_ATTR	6	(step)
	54	LOAD_CONST	0	(None)
	56	COMPARE_OP	9	(is not)
	58	POP_JUMP_IF_FALSE	66	
	60	LOAD_FAST	2	(args)
	62	LOAD_ATTR	6	(step)
	64	JUMP_FORWARD	2	(to 68)
>>	66	LOAD_CONST	2	(1)
>>	68	LOAD_FAST	1	(index)
	70	LOAD_ATTR	6	(step)
	72	LOAD_CONST	0	(None)
	74	COMPARE_OP	9	(is not)
	76	POP_JUMP_IF_FALSE	84	
	78	LOAD_FAST	1	(index)
	80	LOAD_ATTR	6	(step)
	82	JUMP_FORWARD	2	(to 86)
>>	84	LOAD_CONST	2	(1)
>>	86	BUILD_TUPLE	4	
	88	UNPACK_SEQUENCE	4	
	90	STORE_FAST	3	(start)
	92	STORE_FAST	4	(stop)
	94	STORE_FAST	5	(step)
	96	STORE_FAST	6	(ind_step)
52	98	LOAD_FAST	0	(self)
	100	LOAD_METHOD	7	(__class__)
	102	LOAD_FAST	1	(index)
	104	LOAD_ATTR	4	(start)
	106	LOAD_CONST	0	(None)
	108	COMPARE_OP	8	(is)
	110	POP_JUMP_IF_FALSE	132	

	112	LOAD_FAST	6 (ind_step)
	114	LOAD_CONST	1 (0)
	116	COMPARE_OP	5 (>=)
	118	POP_JUMP_IF_FALSE	124
	120	LOAD_FAST	3 (start)
	122	JUMP_ABSOLUTE	168
>>	124	LOAD_FAST	4 (stop)
	126	LOAD_FAST	5 (step)
	128	BINARY_SUBTRACT	
	130	JUMP_FORWARD	36 (to 168)
>>	132	LOAD_FAST	1 (index)
	134	LOAD_ATTR	4 (start)
	136	LOAD_CONST	1 (0)
	138	COMPARE_OP	5 (>=)
	140	POP_JUMP_IF_FALSE	156
	142	LOAD_FAST	3 (start)
	144	LOAD_FAST	5 (step)
	146	LOAD_FAST	1 (index)
	148	LOAD_ATTR	4 (start)
	150	BINARY_MULTIPLY	
	152	BINARY_ADD	
	154	JUMP_FORWARD	12 (to 168)
>>	156	LOAD_FAST	4 (stop)
	158	LOAD_FAST	5 (step)
	160	LOAD_FAST	1 (index)
	162	LOAD_ATTR	4 (start)
	164	BINARY_MULTIPLY	
	166	BINARY_ADD	
>>	168	LOAD_FAST	1 (index)
	170	LOAD_ATTR	5 (stop)
	172	LOAD_CONST	0 (None)
	174	COMPARE_OP	8 (is)
	176	POP_JUMP_IF_FALSE	198
	178	LOAD_FAST	6 (ind_step)
	180	LOAD_CONST	1 (0)
	182	COMPARE_OP	5 (>=)
	184	POP_JUMP_IF_FALSE	190
	186	LOAD_FAST	4 (stop)

	188 JUMP_ABSOLUTE	234
>>	190 LOAD_FAST	3 (start)
	192 LOAD_FAST	5 (step)
	194 BINARY_SUBTRACT	
	196 JUMP_FORWARD	36 (to 234)
>>	198 LOAD_FAST	1 (index)
	200 LOAD_ATTR	5 (stop)
	202 LOAD_CONST	1 (0)
	204 COMPARE_OP	5 (>=)
	206 POP_JUMP_IF_FALSE	222
	208 LOAD_FAST	3 (start)
	210 LOAD_FAST	5 (step)
	212 LOAD_FAST	1 (index)
	214 LOAD_ATTR	5 (stop)
	216 BINARY_MULTIPLY	
	218 BINARY_ADD	
	220 JUMP_FORWARD	12 (to 234)
>>	222 LOAD_FAST	4 (stop)
	224 LOAD_FAST	5 (step)
	226 LOAD_FAST	1 (index)
	228 LOAD_ATTR	5 (stop)
	230 BINARY_MULTIPLY	
	232 BINARY_ADD	
>>	234 LOAD_FAST	1 (index)
	236 LOAD_ATTR	6 (step)
	238 LOAD_CONST	0 (None)
	240 COMPARE_OP	8 (is)
	242 POP_JUMP_IF_FALSE	248
	244 LOAD_FAST	5 (step)
	246 JUMP_FORWARD	8 (to 256)
>>	248 LOAD_FAST	5 (step)
	250 LOAD_FAST	1 (index)
	252 LOAD_ATTR	6 (step)
	254 BINARY_MULTIPLY	
>>	256 CALL_METHOD	3
	258 RETURN_VALUE	
54	>> 260 LOAD_FAST	0 (self)

262	LOAD_ATTR	8 (_xrange)
264	LOAD_FAST	1 (index)
266	BINARY_SUBSCR	
268	RETURN_VALUE	
270	LOAD_CONST	0 (None)
272	RETURN_VALUE	

Disassembly of <code object count at 0x7f6cf328c9d0, file "<dis>", line 55>:

57	0	LOAD_GLOBAL	0 (_coconut_int)
	2	LOAD_FAST	1 (elem)
	4	LOAD_FAST	0 (self)
	6	LOAD_ATTR	1 (_xrange)
	8	COMPARE_OP	6 (in)
	10	CALL_FUNCTION	1
	12	RETURN_VALUE	

Disassembly of <code object index at 0x7f6cf328ca80, file "<dis>", line 58>:

60	0	LOAD_FAST	1 (elem)	
	2	LOAD_FAST	0 (self)	
	4	LOAD_ATTR	0 (_xrange)	
	6	COMPARE_OP	7 (not in)	
	8	POP_JUMP_IF_FALSE	30	
	10	LOAD_GLOBAL	1 (_coconut)	
	12	LOAD_METHOD	2 (ValueError)	
	14	LOAD_GLOBAL	1 (_coconut)	
	16	LOAD_METHOD	3 (repr)	
	18	LOAD_FAST	1 (elem)	
	20	CALL_METHOD	1	
	22	LOAD_CONST	1 (' is not in range')	
	24	BINARY_ADD		
	26	CALL_METHOD	1	
	28	RAISE_VARARGS	1	
61	>>	30	LOAD_FAST	0 (self)
		32	LOAD_ATTR	0 (_xrange)
		34	LOAD_METHOD	4 (__reduce_ex__)
		36	LOAD_CONST	2 (2)
		38	CALL_METHOD	1

	40	LOAD_CONST	3 (1)
	42	BINARY_SUBSCR	
	44	UNPACK_SEQUENCE	3
	46	STORE_FAST	2 (start)
	48	STORE_FAST	3 (_)
	50	STORE_FAST	4 (step)
62	52	LOAD_FAST	1 (elem)
	54	LOAD_FAST	2 (start)
	56	BINARY_SUBTRACT	
	58	LOAD_FAST	4 (step)
	60	BINARY_FLOOR_DIVIDE	
	62	RETURN_VALUE	

Disassembly of <code object __repr__ at 0x7f6cf328cb30, file "<dis>", line 63>:

64	0	LOAD_GLOBAL	0 (_coconut)
	2	LOAD_METHOD	1 (repr)
	4	LOAD_FAST	0 (self)
	6	LOAD_ATTR	2 (_xrange)
	8	CALL_METHOD	1
	10	LOAD_CONST	1 (1)
	12	LOAD_CONST	0 (None)
	14	BUILD_SLICE	2
	16	BINARY_SUBSCR	
	18	RETURN_VALUE	

Disassembly of <code object _args at 0x7f6cf328cbe0, file "<dis>", line 65>:

67	0	LOAD_FAST	0 (self)
	2	LOAD_ATTR	0 (_xrange)
	4	LOAD_METHOD	1 (__reduce__)
	6	CALL_METHOD	0
	8	LOAD_CONST	1 (1)
	10	BINARY_SUBSCR	
	12	RETURN_VALUE	

Disassembly of <code object __reduce_ex__ at 0x7f6cf328cc90, file "<dis>", line 68>:

69	0	LOAD_FAST	0 (self)
	2	LOAD_ATTR	0 (__class__)

4	LOAD_FAST	0 (self)
6	LOAD_ATTR	1 (_xrange)
8	LOAD_METHOD	2 (__reduce_ex__)
10	LOAD_FAST	1 (protocol)
12	CALL_METHOD	1
14	LOAD_CONST	1 (1)
16	BINARY_SUBSCR	
18	BUILD_TUPLE	2
20	RETURN_VALUE	

Disassembly of <code object __reduce__ at 0x7f6cf328cd40, file "<dis>", line 70>:

71	0	LOAD_FAST	0 (self)
	2	LOAD_METHOD	0 (__reduce_ex__)
	4	LOAD_GLOBAL	1 (_coconut)
	6	LOAD_ATTR	2 (pickle)
	8	LOAD_ATTR	3 (DEFAULT_PROTOCOL)
10		CALL_METHOD	1
12		RETURN_VALUE	

Disassembly of <code object __hash__ at 0x7f6cf328cdf0, file "<dis>", line 72>:

73	0	LOAD_GLOBAL	0 (_coconut)
	2	LOAD_METHOD	1 (hash)
	4	LOAD_FAST	0 (self)
	6	LOAD_ATTR	2 (_args)
	8	CALL_METHOD	1
10		RETURN_VALUE	

Disassembly of <code object __copy__ at 0x7f6cf328cea0, file "<dis>", line 74>:

75	0	LOAD_FAST	0 (self)
	2	LOAD_ATTR	0 (__class__)
	4	LOAD_FAST	0 (self)
	6	LOAD_ATTR	1 (_args)
	8	CALL_FUNCTION_EX	0
10		RETURN_VALUE	

Disassembly of <code object __eq__ at 0x7f6cf328cf50, file "<dis>", line 76>:

77	0	LOAD_GLOBAL	0 (_coconut)
	2	LOAD_METHOD	1 (isinstance)

	4	LOAD_FAST	1	(other)
	6	LOAD_FAST	0	(self)
	8	LOAD_ATTR	2	(__class__)
	10	CALL_METHOD	2	
	12	JUMP_IF_FALSE_OR_POP	24	
	14	LOAD_FAST	0	(self)
	16	LOAD_ATTR	3	(_args)
	18	LOAD_FAST	1	(other)
	20	LOAD_ATTR	3	(_args)
	22	COMPARE_OP	2	(==)
>>	24	RETURN_VALUE		

Disassembly of <code object print at 0x7f6cf32993a0, file "<dis>", line 81>:

83	0	LOAD_FAST	1	(kwargs)	
	2	LOAD_METHOD	0	(get)	
	4	LOAD_CONST	1	('file')	
	6	LOAD_GLOBAL	1	(_coconut_sys)	
	8	LOAD_ATTR	2	(stdout)	
	10	CALL_METHOD	2		
	12	STORE_DEREF	0	(file)	
84	14	LOAD_FAST	1	(kwargs)	
	16	LOAD_METHOD	0	(get)	
	18	LOAD_CONST	2	('flush')	
	20	LOAD_CONST	3	(False)	
	22	CALL_METHOD	2		
	24	STORE_FAST	2	(flush)	
85	26	LOAD_CONST	2	('flush')	
	28	LOAD_FAST	1	(kwargs)	
	30	COMPARE_OP	6	(in)	
	32	POP_JUMP_IF_FALSE	40		
86	34	LOAD_FAST	1	(kwargs)	
	36	LOAD_CONST	2	('flush')	
	38	DELETE_SUBSCR			
87	>>	40	LOAD_GLOBAL	3	(_coconut)

		42 LOAD_METHOD	4 (hasattr)
		44 LOAD_DEREF	0 (file)
		46 LOAD_CONST	4 ('encoding')
		48 CALL_METHOD	2
		50 POP_JUMP_IF_FALSE	88
		52 LOAD_DEREF	0 (file)
		54 LOAD_ATTR	5 (encoding)
		56 LOAD_CONST	0 (None)
		58 COMPARE_OP	9 (is not)
		60 POP_JUMP_IF_FALSE	88
88		62 LOAD_GLOBAL	6 (_coconut_print)
		64 LOAD_CLOSURE	0 (file)
		66 BUILD_TUPLE	1
		68 LOAD_CONST	5 (<code object <genexpr> at 0x7f6cf32991
		70 LOAD_CONST	6 ('print.<locals>.<genexpr>')
		72 MAKE_FUNCTION	8 (closure)
		74 LOAD_FAST	0 (args)
		76 GET_ITER	
		78 CALL_FUNCTION	1
		80 LOAD_FAST	1 (kwargs)
		82 CALL_FUNCTION_EX	1
		84 POP_TOP	
		86 JUMP_FORWARD	20 (to 108)
90	>>	88 LOAD_GLOBAL	6 (_coconut_print)
		90 LOAD_CONST	7 (<code object <genexpr> at 0x7f6cf32992
		92 LOAD_CONST	6 ('print.<locals>.<genexpr>')
		94 MAKE_FUNCTION	0
		96 LOAD_FAST	0 (args)
		98 GET_ITER	
		100 CALL_FUNCTION	1
		102 LOAD_FAST	1 (kwargs)
		104 CALL_FUNCTION_EX	1
		106 POP_TOP	
91	>>	108 LOAD_FAST	2 (flush)
		110 POP_JUMP_IF_FALSE	120


```

92      112 LOAD_DEREF          0 (file)
      114 LOAD_METHOD          7 (flush)
      116 CALL_METHOD           0
      118 POP_TOP
>>    120 LOAD_CONST           0 (None)
      122 RETURN_VALUE

```

Disassembly of <code object <genexpr> at 0x7f6cf3299190, file "<dis>", line 88>:

```

88      0 LOAD_FAST              0 (.0)
>>     2 FOR_ITER              22 (to 26)
      4 STORE_FAST              1 (x)
      6 LOAD_GLOBAL             0 (_coconut_unicode)
      8 LOAD_FAST                1 (x)
     10 CALL_FUNCTION            1
     12 LOAD_METHOD              1 (encode)
     14 LOAD_DEREF               0 (file)
     16 LOAD_ATTR                2 (encoding)
     18 CALL_METHOD              1
     20 YIELD_VALUE
     22 POP_TOP
     24 JUMP_ABSOLUTE            2
>>     26 LOAD_CONST             0 (None)
     28 RETURN_VALUE

```

Disassembly of <code object <genexpr> at 0x7f6cf3299240, file "<dis>", line 90>:

```

90      0 LOAD_FAST              0 (.0)
>>     2 FOR_ITER              18 (to 22)
      4 STORE_FAST              1 (x)
      6 LOAD_GLOBAL             0 (_coconut_unicode)
      8 LOAD_FAST                1 (x)
     10 CALL_FUNCTION            1
     12 LOAD_METHOD              1 (encode)
     14 CALL_METHOD              0
     16 YIELD_VALUE
     18 POP_TOP
     20 JUMP_ABSOLUTE            2
>>     22 LOAD_CONST             0 (None)

```

24 RETURN_VALUE

Disassembly of <code object input at 0x7f6cf32992f0, file "<dis>", line 93>:

```

95      0 LOAD_GLOBAL          0 (_coconut)
      2 LOAD_METHOD              1 (hasattr)
      4 LOAD_GLOBAL          2 (_coconut_sys)
      6 LOAD_ATTR              3 (stdout)
      8 LOAD_CONST            1 ('encoding')
     10 CALL_METHOD            2
     12 POP_JUMP_IF_FALSE      46
     14 LOAD_GLOBAL          2 (_coconut_sys)
     16 LOAD_ATTR              3 (stdout)
     18 LOAD_ATTR              4 (encoding)
     20 LOAD_CONST            0 (None)
     22 COMPARE_OP            9 (is not)
     24 POP_JUMP_IF_FALSE      46

96      26 LOAD_GLOBAL          5 (_coconut_raw_input)
     28 LOAD_FAST              0 (args)
     30 LOAD_FAST              1 (kwargs)
     32 CALL_FUNCTION_EX       1
     34 LOAD_METHOD            6 (decode)
     36 LOAD_GLOBAL          2 (_coconut_sys)
     38 LOAD_ATTR              3 (stdout)
     40 LOAD_ATTR              4 (encoding)
     42 CALL_METHOD            1
     44 RETURN_VALUE

97      >> 46 LOAD_GLOBAL          5 (_coconut_raw_input)
      48 LOAD_FAST              0 (args)
      50 LOAD_FAST              1 (kwargs)
      52 CALL_FUNCTION_EX       1
      54 LOAD_METHOD            6 (decode)
      56 CALL_METHOD            0
      58 RETURN_VALUE

```

Disassembly of <code object repr at 0x7f6cf3299450, file "<dis>", line 98>:

```

100      0 LOAD_GLOBAL          0 (isinstance)

```

		2 LOAD_FAST	0 (obj)
		4 LOAD_GLOBAL	1 (_coconut_unicode)
		6 CALL_FUNCTION	2
		8 POP_JUMP_IF_FALSE	30
101		10 LOAD_GLOBAL	1 (_coconut_unicode)
		12 LOAD_GLOBAL	2 (_coconut_repr)
		14 LOAD_FAST	0 (obj)
		16 CALL_FUNCTION	1
		18 LOAD_CONST	1 (1)
		20 LOAD_CONST	0 (None)
		22 BUILD_SLICE	2
		24 BINARY_SUBSCR	
		26 CALL_FUNCTION	1
		28 RETURN_VALUE	
102	>>	30 LOAD_GLOBAL	0 (isinstance)
		32 LOAD_FAST	0 (obj)
		34 LOAD_GLOBAL	3 (_coconut_str)
		36 CALL_FUNCTION	2
		38 POP_JUMP_IF_FALSE	56
103		40 LOAD_CONST	2 ('b')
		42 LOAD_GLOBAL	1 (_coconut_unicode)
		44 LOAD_GLOBAL	2 (_coconut_repr)
		46 LOAD_FAST	0 (obj)
		48 CALL_FUNCTION	1
		50 CALL_FUNCTION	1
		52 BINARY_ADD	
		54 RETURN_VALUE	
104	>>	56 LOAD_GLOBAL	1 (_coconut_unicode)
		58 LOAD_GLOBAL	2 (_coconut_repr)
		60 LOAD_FAST	0 (obj)
		62 CALL_FUNCTION	1
		64 CALL_FUNCTION	1
		66 RETURN_VALUE	

```

Disassembly of <code object raw_input at 0x7f6cf3299500, file "<dis>", line 106>:
108      0 LOAD_GLOBAL              0 (_coconut)
        2 LOAD_METHOD                1 (NameError)
        4 LOAD_CONST                  1 ('Coconut uses Python 3 "input" instead
        6 CALL_METHOD              1
        8 RAISE_VARARGS            1
       10 LOAD_CONST                2 (None)
       12 RETURN_VALUE

```

```

Disassembly of <code object xrange at 0x7f6cf32995b0, file "<dis>", line 109>:
111      0 LOAD_GLOBAL              0 (_coconut)
        2 LOAD_METHOD                1 (NameError)
        4 LOAD_CONST                  1 ('Coconut uses Python 3 "range" instead
        6 CALL_METHOD              1
        8 RAISE_VARARGS            1
       10 LOAD_CONST                2 (None)
       12 RETURN_VALUE

```

```

Disassembly of <code object _coconut_new_partial at 0x7f6cf3299660, file "<dis>",
115      0 LOAD_GLOBAL              0 (_coconut_func tools)
        2 LOAD_ATTR                  1 (partial)
        4 LOAD_FAST                  0 (func)
        6 BUILD_TUPLE              1
        8 LOAD_FAST                  1 (args)
       10 LOAD_CONST                  0 (None)
       12 COMPARE_OP                  9 (is not)
       14 POP_JUMP_IF_FALSE          20
       16 LOAD_FAST                  1 (args)
       18 JUMP_FORWARD                2 (to 22)
>>    20 LOAD_CONST                  1 (())
>>    22 BUILD_TUPLE_UNPACK_WITH_CALL    2
       24 LOAD_FAST                  2 (keywords)
       26 LOAD_CONST                  0 (None)
       28 COMPARE_OP                  9 (is not)
       30 POP_JUMP_IF_FALSE          36
       32 LOAD_FAST                  2 (keywords)
       34 JUMP_FORWARD                2 (to 38)
>>    36 BUILD_MAP                  0

```

```
>> 38 CALL_FUNCTION_EX      1
    40 RETURN_VALUE
```

Disassembly of <code object _coconut_reduce_partial at 0x7f6cf3299710, file "<dis>">:

```
118      0 LOAD_GLOBAL          0 (_coconut_new_partial)
      2 LOAD_FAST                0 (self)
      4 LOAD_ATTR              1 (func)
      6 LOAD_FAST                0 (self)
      8 LOAD_ATTR              2 (args)
     10 LOAD_FAST                0 (self)
     12 LOAD_ATTR              3 (keywords)
     14 BUILD_TUPLE            3
     16 BUILD_TUPLE            2
     18 RETURN_VALUE
```

Disassembly of <code object _coconut at 0x7f6cf3299b30, file "<dis>", line 124>:

```
124      0 LOAD_NAME              0 (__name__)
      2 STORE_NAME              1 (__module__)
      4 LOAD_CONST              0 ('_coconut')
      6 STORE_NAME              2 (__qualname__)

125      8 LOAD_CONST              1 (0)
     10 LOAD_CONST              2 (None)
     12 IMPORT_NAME            3 (collections)
     14 STORE_NAME              3 (collections)
     16 LOAD_CONST              1 (0)
     18 LOAD_CONST              2 (None)
     20 IMPORT_NAME            4 (copy)
     22 STORE_NAME              4 (copy)
     24 LOAD_CONST              1 (0)
     26 LOAD_CONST              2 (None)
     28 IMPORT_NAME            5 (functools)
     30 STORE_NAME              5 (functools)
     32 LOAD_CONST              1 (0)
     34 LOAD_CONST              2 (None)
     36 IMPORT_NAME            6 (types)
     38 STORE_NAME              6 (types)
     40 LOAD_CONST              1 (0)
```

	42	LOAD_CONST	2	(None)
	44	IMPORT_NAME	7	(itertools)
	46	STORE_NAME	7	(itertools)
	48	LOAD_CONST	1	(0)
	50	LOAD_CONST	2	(None)
	52	IMPORT_NAME	8	(operator)
	54	STORE_NAME	8	(operator)
	56	LOAD_CONST	1	(0)
	58	LOAD_CONST	2	(None)
	60	IMPORT_NAME	9	(threading)
	62	STORE_NAME	9	(threading)
	64	LOAD_CONST	1	(0)
	66	LOAD_CONST	2	(None)
	68	IMPORT_NAME	10	(weakref)
	70	STORE_NAME	10	(weakref)
	72	LOAD_CONST	1	(0)
	74	LOAD_CONST	2	(None)
	76	IMPORT_NAME	11	(os)
	78	STORE_NAME	11	(os)
	80	LOAD_CONST	1	(0)
	82	LOAD_CONST	2	(None)
	84	IMPORT_NAME	12	(warnings)
	86	STORE_NAME	12	(warnings)
126	88	LOAD_NAME	13	(_coconut_sys)
	90	LOAD_ATTR	14	(version_info)
	92	LOAD_CONST	3	((3, 2))
	94	COMPARE_OP	0	(<)
	96	POP_JUMP_IF_FALSE	142	
127	98	SETUP_FINALLY	22	(to 122)
128	100	LOAD_CONST	1	(0)
	102	LOAD_CONST	4	(('lru_cache',))
	104	IMPORT_NAME	15	(backports.functools_lru_cache)
	106	IMPORT_FROM	16	(lru_cache)
	108	STORE_NAME	16	(lru_cache)
	110	POP_TOP		

129		112 LOAD_NAME	16 (lru_cache)
		114 LOAD_NAME	5 (functools)
		116 STORE_ATTR	16 (lru_cache)
		118 POP_BLOCK	
		120 JUMP_FORWARD	20 (to 142)
130	>>	122 DUP_TOP	
		124 LOAD_NAME	17 (ImportError)
		126 COMPARE_OP	10 (exception match)
		128 POP_JUMP_IF_FALSE	140
		130 POP_TOP	
		132 POP_TOP	
		134 POP_TOP	
		136 POP_EXCEPT	
		138 JUMP_FORWARD	2 (to 142)
	>>	140 END_FINALLY	
131	>>	142 LOAD_NAME	13 (_coconut_sys)
		144 LOAD_ATTR	14 (version_info)
		146 LOAD_CONST	5 ((3,))
		148 COMPARE_OP	0 (<)
		150 POP_JUMP_IF_FALSE	162
132		152 LOAD_CONST	1 (0)
		154 LOAD_CONST	2 (None)
		156 IMPORT_NAME	18 (cPickle)
		158 STORE_NAME	19 (pickle)
		160 JUMP_FORWARD	8 (to 170)
134	>>	162 LOAD_CONST	1 (0)
		164 LOAD_CONST	2 (None)
		166 IMPORT_NAME	19 (pickle)
		168 STORE_NAME	19 (pickle)
135	>>	170 LOAD_NAME	13 (_coconut_sys)
		172 LOAD_ATTR	14 (version_info)
		174 LOAD_CONST	6 ((2, 7))

		176 COMPARE_OP	5 (>=)
		178 POP_JUMP_IF_FALSE	188
136		180 LOAD_NAME	3 (collections)
		182 LOAD_ATTR	20 (OrderedDict)
		184 STORE_NAME	20 (OrderedDict)
		186 JUMP_FORWARD	4 (to 192)
138	>>	188 LOAD_NAME	21 (dict)
		190 STORE_NAME	20 (OrderedDict)
139	>>	192 LOAD_NAME	13 (_coconut_sys)
		194 LOAD_ATTR	14 (version_info)
		196 LOAD_CONST	7 ((3, 3))
		198 COMPARE_OP	0 (<)
		200 POP_JUMP_IF_FALSE	208
140		202 LOAD_NAME	3 (collections)
		204 STORE_NAME	22 (abc)
		206 JUMP_FORWARD	12 (to 220)
142	>>	208 LOAD_CONST	1 (0)
		210 LOAD_CONST	2 (None)
		212 IMPORT_NAME	23 (collections.abc)
		214 IMPORT_FROM	22 (abc)
		216 STORE_NAME	22 (abc)
		218 POP_TOP	
143	>>	220 LOAD_BUILD_CLASS	
		222 LOAD_CONST	8 (<code object typing at 0x7f6cf3299a80,
		224 LOAD_CONST	9 ('typing')
		226 MAKE_FUNCTION	0
		228 LOAD_CONST	9 ('typing')
		230 LOAD_NAME	24 (object)
		232 CALL_FUNCTION	3
		234 STORE_NAME	25 (typing)
147		236 LOAD_NAME	26 (Ellipsis)

238	LOAD_NAME	27	(Exception)
240	LOAD_NAME	28	(AttributeError)
242	LOAD_NAME	17	(ImportError)
244	LOAD_NAME	29	(IndexError)
246	LOAD_NAME	30	(KeyError)
248	LOAD_NAME	31	(NameError)
250	LOAD_NAME	32	(TypeError)
252	LOAD_NAME	33	(ValueError)
254	LOAD_NAME	34	(StopIteration)
256	LOAD_NAME	35	(classmethod)
258	LOAD_NAME	21	(dict)
260	LOAD_NAME	36	(enumerate)
262	LOAD_NAME	37	(filter)
264	LOAD_NAME	38	(float)
266	LOAD_NAME	39	(frozenset)
268	LOAD_NAME	40	(getattr)
270	LOAD_NAME	41	(hasattr)
272	LOAD_NAME	42	(hash)
274	LOAD_NAME	43	(id)
276	LOAD_NAME	44	(int)
278	LOAD_NAME	45	(isinstance)
280	LOAD_NAME	46	(issubclass)
282	LOAD_NAME	47	(iter)
284	LOAD_NAME	48	(len)
286	LOAD_NAME	49	(list)
288	LOAD_NAME	50	(locals)
290	LOAD_NAME	51	(map)
292	LOAD_NAME	52	(min)
294	LOAD_NAME	53	(max)
296	LOAD_NAME	54	(next)
298	LOAD_NAME	24	(object)
300	LOAD_NAME	55	(property)
302	LOAD_NAME	56	(range)
304	LOAD_NAME	57	(reversed)
306	LOAD_NAME	58	(set)
308	LOAD_NAME	59	(slice)
310	LOAD_NAME	60	(str)
312	LOAD_NAME	61	(sum)

314	LOAD_NAME	62	(super)
316	LOAD_NAME	63	(tuple)
318	LOAD_NAME	64	(type)
320	LOAD_NAME	65	(zip)
322	LOAD_NAME	66	(staticmethod)
324	LOAD_NAME	67	(repr)
326	CALL_FUNCTION	1	
328	LOAD_NAME	68	(bytearray)
330	BUILD_TUPLE	45	
332	UNPACK_SEQUENCE	45	
334	STORE_NAME	26	(Ellipsis)
336	STORE_NAME	27	(Exception)
338	STORE_NAME	28	(AttributeError)
340	STORE_NAME	17	(ImportError)
342	STORE_NAME	29	(IndexError)
344	STORE_NAME	30	(KeyError)
346	STORE_NAME	31	(NameError)
348	STORE_NAME	32	(TypeError)
350	STORE_NAME	33	(ValueError)
352	STORE_NAME	34	(StopIteration)
354	STORE_NAME	35	(classmethod)
356	STORE_NAME	21	(dict)
358	STORE_NAME	36	(enumerate)
360	STORE_NAME	37	(filter)
362	STORE_NAME	38	(float)
364	STORE_NAME	39	(frozenset)
366	STORE_NAME	40	(getattr)
368	STORE_NAME	41	(hasattr)
370	STORE_NAME	42	(hash)
372	STORE_NAME	43	(id)
374	STORE_NAME	44	(int)
376	STORE_NAME	45	(isinstance)
378	STORE_NAME	46	(issubclass)
380	STORE_NAME	47	(iter)
382	STORE_NAME	48	(len)
384	STORE_NAME	49	(list)
386	STORE_NAME	50	(locals)
388	STORE_NAME	51	(map)

390	STORE_NAME	52 (min)
392	STORE_NAME	53 (max)
394	STORE_NAME	54 (next)
396	STORE_NAME	24 (object)
398	STORE_NAME	55 (property)
400	STORE_NAME	56 (range)
402	STORE_NAME	57 (reversed)
404	STORE_NAME	58 (set)
406	STORE_NAME	59 (slice)
408	STORE_NAME	60 (str)
410	STORE_NAME	61 (sum)
412	STORE_NAME	62 (super)
414	STORE_NAME	63 (tuple)
416	STORE_NAME	64 (type)
418	STORE_NAME	65 (zip)
420	STORE_NAME	67 (repr)
422	STORE_NAME	68 (bytearray)
424	LOAD_CONST	2 (None)
426	RETURN_VALUE	

Disassembly of <code object typing at 0x7f6cf3299a80, file "<dis>", line 143>:

143	0	LOAD_NAME	0	(__name__)
	2	STORE_NAME	1	(__module__)
	4	LOAD_CONST	0	('_coconut.typing')
	6	STORE_NAME	2	(__qualname__)
144	8	LOAD_NAME	3	(staticmethod)
145	10	LOAD_CONST	1	(<code object NamedTuple at 0x7f6cf32999d0, file "<dis>", line 144>)
	12	LOAD_CONST	2	('_coconut.typing.NamedTuple')
	14	MAKE_FUNCTION	0	
	16	CALL_FUNCTION	1	
	18	STORE_NAME	4	(NamedTuple)
	20	LOAD_CONST	3	(None)
	22	RETURN_VALUE		

Disassembly of <code object NamedTuple at 0x7f6cf32999d0, file "<dis>", line 144>:

146	0	LOAD_GLOBAL	0	(_coconut)
-----	---	-------------	---	------------

2	LOAD_ATTR	1 (collections)
4	LOAD_METHOD	2 (namedtuple)
6	LOAD_FAST	0 (name)
8	LOAD_CONST	1 (<code object <listcomp> at 0x7f6cf3299920, file "<dis>", line 146>)
10	LOAD_CONST	2 ('_coconut.typing.NamedTuple.<locals>.<locals>')
12	MAKE_FUNCTION	0
14	LOAD_FAST	1 (fields)
16	GET_ITER	
18	CALL_FUNCTION	1
20	CALL_METHOD	2
22	RETURN_VALUE	

Disassembly of <code object <listcomp> at 0x7f6cf3299920, file "<dis>", line 146>:

146	0	BUILD_LIST	0
	2	LOAD_FAST	0 (.0)
>>	4	FOR_ITER	12 (to 18)
	6	UNPACK_SEQUENCE	2
	8	STORE_FAST	1 (x)
	10	STORE_FAST	2 (t)
	12	LOAD_FAST	1 (x)
	14	LIST_APPEND	2
	16	JUMP_ABSOLUTE	4
>>	18	RETURN_VALUE	

Disassembly of <code object MatchError at 0x7f6cf3299be0, file "<dis>", line 149>:

149	0	LOAD_NAME	0 (__name__)
	2	STORE_NAME	1 (__module__)
	4	LOAD_CONST	0 ('MatchError')
	6	STORE_NAME	2 (__qualname__)
150	8	LOAD_CONST	1 ('Pattern-matching error. Has attribute')
	10	STORE_NAME	3 (__doc__)
151	12	LOAD_CONST	2 (('pattern', 'value'))
	14	STORE_NAME	4 (__slots__)
	16	LOAD_CONST	3 (None)
	18	RETURN_VALUE	

Disassembly of <code object _coconut_tail_call at 0x7f6cf3299d40, file "<dis>", line 152>:

```

152      0 LOAD_NAME          0 (__name__)
      2 STORE_NAME            1 (__module__)
      4 LOAD_CONST           0 ('_coconut_tail_call')
      6 STORE_NAME          2 (__qualname__)

153      8 LOAD_CONST           1 (('func', 'args', 'kwargs'))
     10 STORE_NAME          3 (__slots__)

154     12 LOAD_CONST           2 (<code object __init__ at 0x7f6cf3299c90>)
     14 LOAD_CONST           3 ('_coconut_tail_call.__init__')
     16 MAKE_FUNCTION        0
     18 STORE_NAME          4 (__init__)
     20 LOAD_CONST           4 (None)
     22 RETURN_VALUE

```

Disassembly of <code object __init__ at 0x7f6cf3299c90, file "<dis>", line 154>:

```

155      0 LOAD_FAST            1 (func)
      2 LOAD_FAST            2 (args)
      4 LOAD_FAST            3 (kwargs)
      6 ROT_THREE
      8 ROT_TWO
     10 LOAD_FAST            0 (self)
     12 STORE_ATTR           0 (func)
     14 LOAD_FAST            0 (self)
     16 STORE_ATTR           1 (args)
     18 LOAD_FAST            0 (self)
     20 STORE_ATTR           2 (kwargs)
     22 LOAD_CONST           0 (None)
     24 RETURN_VALUE

```

Disassembly of <code object _coconut_tco at 0x7f6cf329c030, file "<dis>", line 157>:

```

158      0 LOAD_GLOBAL          0 (_coconut)
      2 LOAD_ATTR             1 (functools)
      4 LOAD_METHOD          2 (wraps)
      6 LOAD_DEREF           0 (func)
      8 CALL_METHOD         1

```

159	10 LOAD_CLOSURE	0 (func)
	12 BUILD_TUPLE	1
	14 LOAD_CONST	1 (<code object tail_call_optimized_func
	16 LOAD_CONST	2 ('_coconut_tco.<locals>.tail_call_optim
	18 MAKE_FUNCTION	8 (closure)
	20 CALL_FUNCTION	1
	22 STORE_FAST	1 (tail_call_optimized_func)
169	24 LOAD_DEREF	0 (func)
	26 LOAD_FAST	1 (tail_call_optimized_func)
	28 STORE_ATTR	3 (_coconut_tco_func)
170	30 LOAD_GLOBAL	0 (_coconut)
	32 LOAD_METHOD	4 (getattr)
	34 LOAD_DEREF	0 (func)
	36 LOAD_CONST	3 ('__module__')
	38 LOAD_CONST	0 (None)
	40 CALL_METHOD	3
	42 LOAD_FAST	1 (tail_call_optimized_func)
	44 STORE_ATTR	5 (__module__)
171	46 LOAD_GLOBAL	0 (_coconut)
	48 LOAD_METHOD	4 (getattr)
	50 LOAD_DEREF	0 (func)
	52 LOAD_CONST	4 ('__name__')
	54 LOAD_CONST	5 ('<coconut tco function (pass --no-tco
	56 CALL_METHOD	3
	58 LOAD_FAST	1 (tail_call_optimized_func)
	60 STORE_ATTR	6 (__name__)
172	62 LOAD_GLOBAL	0 (_coconut)
	64 LOAD_METHOD	4 (getattr)
	66 LOAD_DEREF	0 (func)
	68 LOAD_CONST	6 ('__qualname__')
	70 LOAD_FAST	1 (tail_call_optimized_func)
	72 LOAD_ATTR	6 (__name__)
	74 CALL_METHOD	3
	76 LOAD_FAST	1 (tail_call_optimized_func)

	78	STORE_ATTR	7	(__qualname__)
173	80	LOAD_GLOBAL	0	(_coconut)
	82	LOAD_ATTR	8	(weakref)
	84	LOAD_METHOD	9	(ref)
	86	LOAD_FAST	1	(tail_call_optimized_func)
	88	CALL_METHOD	1	
	90	LOAD_GLOBAL	10	(_coconut_tco_func_dict)
	92	LOAD_GLOBAL	0	(_coconut)
	94	LOAD_METHOD	11	(id)
	96	LOAD_FAST	1	(tail_call_optimized_func)
	98	CALL_METHOD	1	
	100	STORE_SUBSCR		
174	102	LOAD_FAST	1	(tail_call_optimized_func)
	104	RETURN_VALUE		

Disassembly of <code object tail_call_optimized_func at 0x7f6cf3299f50, file "<dis

160		0	LOAD_DEREF	0	(func)
		2	STORE_FAST	2	(call_func)
162	>>	4	LOAD_GLOBAL	0	(_coconut_tco_func_dict)
		6	LOAD_METHOD	1	(get)
		8	LOAD_GLOBAL	2	(_coconut)
		10	LOAD_METHOD	3	(id)
		12	LOAD_FAST	2	(call_func)
		14	CALL_METHOD	1	
		16	CALL_METHOD	1	
		18	STORE_FAST	3	(wkref)
163		20	LOAD_FAST	3	(wkref)
		22	LOAD_CONST	0	(None)
		24	COMPARE_OP	9	(is not)
		26	POP_JUMP_IF_FALSE	38	
		28	LOAD_FAST	3	(wkref)
		30	CALL_FUNCTION	0	
		32	LOAD_FAST	2	(call_func)
		34	COMPARE_OP	8	(is)

		36 POP_JUMP_IF_TRUE	50
>>		38 LOAD_GLOBAL	2 (_coconut)
		40 LOAD_METHOD	4 (isinstance)
		42 LOAD_FAST	2 (call_func)
		44 LOAD_GLOBAL	5 (_coconut_base_pattern_func)
		46 CALL_METHOD	2
		48 POP_JUMP_IF_FALSE	56
164	>>	50 LOAD_FAST	2 (call_func)
		52 LOAD_ATTR	6 (_coconut_tco_func)
		54 STORE_FAST	2 (call_func)
165	>>	56 LOAD_FAST	2 (call_func)
		58 LOAD_FAST	0 (args)
		60 LOAD_FAST	1 (kwargs)
		62 CALL_FUNCTION_EX	1
		64 STORE_FAST	4 (result)
166		66 LOAD_GLOBAL	4 (isinstance)
		68 LOAD_FAST	4 (result)
		70 LOAD_GLOBAL	7 (_coconut_tail_call)
		72 CALL_FUNCTION	2
		74 POP_JUMP_IF_TRUE	80
167		76 LOAD_FAST	4 (result)
		78 RETURN_VALUE	
168	>>	80 LOAD_FAST	4 (result)
		82 LOAD_ATTR	8 (func)
		84 LOAD_FAST	4 (result)
		86 LOAD_ATTR	9 (args)
		88 LOAD_FAST	4 (result)
		90 LOAD_ATTR	10 (kwargs)
		92 ROT_THREE	
		94 ROT_TWO	
		96 STORE_FAST	2 (call_func)
		98 STORE_FAST	0 (args)
		100 STORE_FAST	1 (kwargs)

102	JUMP_ABSOLUTE	4
104	LOAD_CONST	0 (None)
106	RETURN_VALUE	

Disassembly of <code object _coconut_igetitem at 0x7f6cf3299ea0, file "<dis>", line 176

176	0	LOAD_GLOBAL	0 (isinstance)
	2	LOAD_FAST	0 (iterable)
	4	LOAD_GLOBAL	1 (_coconut_reversed)
	6	LOAD_GLOBAL	2 (_coconut_map)
	8	LOAD_GLOBAL	3 (_coconut)
	10	LOAD_ATTR	4 (zip)
	12	LOAD_GLOBAL	5 (_coconut_enumerate)
	14	LOAD_GLOBAL	6 (_coconut_count)
	16	LOAD_GLOBAL	3 (_coconut)
	18	LOAD_ATTR	7 (abc)
	20	LOAD_ATTR	8 (Sequence)
	22	BUILD_TUPLE	6
	24	CALL_FUNCTION	2
	26	POP_JUMP_IF_FALSE	36

177	28	LOAD_FAST	0 (iterable)
	30	LOAD_FAST	1 (index)
	32	BINARY_SUBSCR	
	34	RETURN_VALUE	

178	>>	36	LOAD_GLOBAL	3 (_coconut)
		38	LOAD_METHOD	0 (isinstance)
		40	LOAD_FAST	1 (index)
		42	LOAD_GLOBAL	3 (_coconut)
		44	LOAD_ATTR	9 (slice)
		46	CALL_METHOD	2
		48	POP_JUMP_IF_TRUE	106

179	50	LOAD_FAST	1 (index)
	52	LOAD_CONST	1 (0)
	54	COMPARE_OP	0 (<)
	56	POP_JUMP_IF_FALSE	80

180		58 LOAD_GLOBAL	3 (_coconut)
		60 LOAD_ATTR	10 (collections)
		62 LOAD_ATTR	11 (deque)
		64 LOAD_FAST	0 (iterable)
		66 LOAD_FAST	1 (index)
		68 UNARY_NEGATIVE	
		70 LOAD_CONST	2 (('maxlen',))
		72 CALL_FUNCTION_KW	2
		74 LOAD_CONST	1 (0)
		76 BINARY_SUBSCR	
		78 RETURN_VALUE	
181	>>	80 LOAD_GLOBAL	3 (_coconut)
		82 LOAD_METHOD	12 (next)
		84 LOAD_GLOBAL	3 (_coconut)
		86 LOAD_ATTR	13 (itertools)
		88 LOAD_METHOD	14 (islice)
		90 LOAD_FAST	0 (iterable)
		92 LOAD_FAST	1 (index)
		94 LOAD_FAST	1 (index)
		96 LOAD_CONST	3 (1)
		98 BINARY_ADD	
		100 CALL_METHOD	3
		102 CALL_METHOD	1
		104 RETURN_VALUE	
182	>>	106 LOAD_FAST	1 (index)
		108 LOAD_ATTR	15 (start)
		110 LOAD_CONST	0 (None)
		112 COMPARE_OP	9 (is not)
		114 POP_JUMP_IF_FALSE	216
		116 LOAD_FAST	1 (index)
		118 LOAD_ATTR	15 (start)
		120 LOAD_CONST	1 (0)
		122 COMPARE_OP	0 (<)
		124 POP_JUMP_IF_FALSE	216
		126 LOAD_FAST	1 (index)
		128 LOAD_ATTR	16 (stop)

	130	LOAD_CONST	0 (None)
	132	COMPARE_OP	8 (is)
	134	POP_JUMP_IF_TRUE	146
	136	LOAD_FAST	1 (index)
	138	LOAD_ATTR	16 (stop)
	140	LOAD_CONST	1 (0)
	142	COMPARE_OP	0 (<)
	144	POP_JUMP_IF_FALSE	216
>>	146	LOAD_FAST	1 (index)
	148	LOAD_ATTR	17 (step)
	150	LOAD_CONST	0 (None)
	152	COMPARE_OP	8 (is)
	154	POP_JUMP_IF_FALSE	216
183	156	LOAD_GLOBAL	3 (_coconut)
	158	LOAD_ATTR	10 (collections)
	160	LOAD_ATTR	11 (deque)
	162	LOAD_FAST	0 (iterable)
	164	LOAD_FAST	1 (index)
	166	LOAD_ATTR	15 (start)
	168	UNARY_NEGATIVE	
	170	LOAD_CONST	2 (('maxlen',))
	172	CALL_FUNCTION_KW	2
	174	STORE_FAST	2 (queue)
184	176	LOAD_FAST	1 (index)
	178	LOAD_ATTR	16 (stop)
	180	LOAD_CONST	0 (None)
	182	COMPARE_OP	9 (is not)
	184	POP_JUMP_IF_FALSE	212
185	186	LOAD_GLOBAL	3 (_coconut)
	188	LOAD_METHOD	18 (list)
	190	LOAD_FAST	2 (queue)
	192	CALL_METHOD	1
	194	LOAD_CONST	0 (None)
	196	LOAD_FAST	1 (index)
	198	LOAD_ATTR	16 (stop)

		200 LOAD_FAST	1 (index)
		202 LOAD_ATTR	15 (start)
		204 BINARY_SUBTRACT	
		206 BUILD_SLICE	2
		208 BINARY_SUBSCR	
		210 STORE_FAST	2 (queue)
186	>>	212 LOAD_FAST	2 (queue)
		214 RETURN_VALUE	
187	>>	216 LOAD_FAST	1 (index)
		218 LOAD_ATTR	15 (start)
		220 LOAD_CONST	0 (None)
		222 COMPARE_OP	9 (is not)
		224 POP_JUMP_IF_FALSE	238
		226 LOAD_FAST	1 (index)
		228 LOAD_ATTR	15 (start)
		230 LOAD_CONST	1 (0)
		232 COMPARE_OP	0 (<)
		234 EXTENDED_ARG	1
		236 POP_JUMP_IF_TRUE	286
	>>	238 LOAD_FAST	1 (index)
		240 LOAD_ATTR	16 (stop)
		242 LOAD_CONST	0 (None)
		244 COMPARE_OP	9 (is not)
		246 EXTENDED_ARG	1
		248 POP_JUMP_IF_FALSE	262
		250 LOAD_FAST	1 (index)
		252 LOAD_ATTR	16 (stop)
		254 LOAD_CONST	1 (0)
		256 COMPARE_OP	0 (<)
		258 EXTENDED_ARG	1
		260 POP_JUMP_IF_TRUE	286
	>>	262 LOAD_FAST	1 (index)
		264 LOAD_ATTR	17 (step)
		266 LOAD_CONST	0 (None)
		268 COMPARE_OP	9 (is not)
		270 EXTENDED_ARG	1

		272 POP_JUMP_IF_FALSE	300
		274 LOAD_FAST	1 (index)
		276 LOAD_ATTR	17 (step)
		278 LOAD_CONST	1 (0)
		280 COMPARE_OP	0 (<)
		282 EXTENDED_ARG	1
		284 POP_JUMP_IF_FALSE	300
188	>>	286 LOAD_GLOBAL	3 (_coconut)
		288 LOAD_METHOD	18 (list)
		290 LOAD_FAST	0 (iterable)
		292 CALL_METHOD	1
		294 LOAD_FAST	1 (index)
		296 BINARY_SUBSCR	
		298 RETURN_VALUE	
189	>>	300 LOAD_GLOBAL	3 (_coconut)
		302 LOAD_ATTR	13 (itertools)
		304 LOAD_METHOD	14 (islice)
		306 LOAD_FAST	0 (iterable)
		308 LOAD_FAST	1 (index)
		310 LOAD_ATTR	15 (start)
		312 LOAD_FAST	1 (index)
		314 LOAD_ATTR	16 (stop)
		316 LOAD_FAST	1 (index)
		318 LOAD_ATTR	17 (step)
		320 CALL_METHOD	4
		322 RETURN_VALUE	

Disassembly of <code object _coconut_base_compose at 0x7f6cf329c500, file "<dis>".

190		0 LOAD_NAME	0 (__name__)
		2 STORE_NAME	1 (__module__)
		4 LOAD_CONST	0 ('_coconut_base_compose')
		6 STORE_NAME	2 (__qualname__)
191		8 LOAD_CONST	1 (('func', 'funcstars'))
		10 STORE_NAME	3 (__slots__)

192	12 LOAD_CONST	2 (<code object __init__ at 0x7f6cf329c0e0
	14 LOAD_CONST	3 ('_coconut_base_compose.__init__')
	16 MAKE_FUNCTION	0
	18 STORE_NAME	4 (__init__)
201	20 LOAD_CONST	4 (<code object __call__ at 0x7f6cf329c2d0
	22 LOAD_CONST	5 ('_coconut_base_compose.__call__')
	24 MAKE_FUNCTION	0
	26 STORE_NAME	5 (__call__)
213	28 LOAD_CONST	6 (<code object __repr__ at 0x7f6cf329c2f0
	30 LOAD_CONST	7 ('_coconut_base_compose.__repr__')
	32 MAKE_FUNCTION	0
	34 STORE_NAME	6 (__repr__)
215	36 LOAD_CONST	8 (<code object __reduce__ at 0x7f6cf329c310
	38 LOAD_CONST	9 ('_coconut_base_compose.__reduce__')
	40 MAKE_FUNCTION	0
	42 STORE_NAME	7 (__reduce__)
217	44 LOAD_CONST	13 ((None,))
	46 LOAD_CONST	11 (<code object __get__ at 0x7f6cf329c450
	48 LOAD_CONST	12 ('_coconut_base_compose.__get__')
	50 MAKE_FUNCTION	1 (defaults)
	52 STORE_NAME	8 (__get__)
	54 LOAD_CONST	10 (None)
	56 RETURN_VALUE	

Disassembly of <code object __init__ at 0x7f6cf329c0e0, file "<dis>", line 192>:

193	0 LOAD_FAST	1 (func)
	2 LOAD_FAST	0 (self)
	4 STORE_ATTR	0 (func)
194	6 BUILD_LIST	0
	8 LOAD_FAST	0 (self)
	10 STORE_ATTR	1 (funcstars)
195	12 LOAD_FAST	2 (funcstars)

	14 GET_ITER	
>>	16 FOR_ITER	72 (to 90)
	18 UNPACK_SEQUENCE	2
	20 STORE_FAST	3 (f)
	22 STORE_FAST	4 (stars)
196	24 LOAD_GLOBAL	2 (_coconut)
	26 LOAD_METHOD	3 (isinstance)
	28 LOAD_FAST	3 (f)
	30 LOAD_GLOBAL	4 (_coconut_base_compose)
	32 CALL_METHOD	2
	34 POP_JUMP_IF_FALSE	72
197	36 LOAD_FAST	0 (self)
	38 LOAD_ATTR	1 (funcstars)
	40 LOAD_METHOD	5 (append)
	42 LOAD_FAST	3 (f)
	44 LOAD_ATTR	0 (func)
	46 LOAD_FAST	4 (stars)
	48 BUILD_TUPLE	2
	50 CALL_METHOD	1
	52 POP_TOP	
198	54 LOAD_FAST	0 (self)
	56 DUP_TOP	
	58 LOAD_ATTR	1 (funcstars)
	60 LOAD_FAST	3 (f)
	62 LOAD_ATTR	1 (funcstars)
	64 INPLACE_ADD	
	66 ROT_TWO	
	68 STORE_ATTR	1 (funcstars)
	70 JUMP_ABSOLUTE	16
200	>> 72 LOAD_FAST	0 (self)
	74 LOAD_ATTR	1 (funcstars)
	76 LOAD_METHOD	5 (append)
	78 LOAD_FAST	3 (f)
	80 LOAD_FAST	4 (stars)

	82 BUILD_TUPLE	2
	84 CALL_METHOD	1
	86 POP_TOP	
	88 JUMP_ABSOLUTE	16
>>	90 LOAD_CONST	0 (None)
	92 RETURN_VALUE	

Disassembly of <code object __call__ at 0x7f6cf329c240, file "<dis>", line 201>:

202	0 LOAD_FAST	0 (self)
	2 LOAD_ATTR	0 (func)
	4 LOAD_FAST	1 (args)
	6 LOAD_FAST	2 (kwargs)
	8 CALL_FUNCTION_EX	1
	10 STORE_FAST	3 (arg)
203	12 LOAD_FAST	0 (self)
	14 LOAD_ATTR	1 (funcstars)
	16 GET_ITER	
>>	18 FOR_ITER	84 (to 104)
	20 UNPACK_SEQUENCE	2
	22 STORE_FAST	4 (f)
	24 STORE_FAST	5 (stars)
204	26 LOAD_FAST	5 (stars)
	28 LOAD_CONST	1 (0)
	30 COMPARE_OP	2 (==)
	32 POP_JUMP_IF_FALSE	44
205	34 LOAD_FAST	4 (f)
	36 LOAD_FAST	3 (arg)
	38 CALL_FUNCTION	1
	40 STORE_FAST	3 (arg)
	42 JUMP_ABSOLUTE	18
206	>> 44 LOAD_FAST	5 (stars)
	46 LOAD_CONST	2 (1)
	48 COMPARE_OP	2 (==)
	50 POP_JUMP_IF_FALSE	62

207		52 LOAD_FAST	4 (f)
		54 LOAD_FAST	3 (arg)
		56 CALL_FUNCTION_EX	0
		58 STORE_FAST	3 (arg)
		60 JUMP_ABSOLUTE	18
208	>>	62 LOAD_FAST	5 (stars)
		64 LOAD_CONST	3 (2)
		66 COMPARE_OP	2 (==)
		68 POP_JUMP_IF_FALSE	82
209		70 LOAD_FAST	4 (f)
		72 BUILD_TUPLE	0
		74 LOAD_FAST	3 (arg)
		76 CALL_FUNCTION_EX	1
		78 STORE_FAST	3 (arg)
		80 JUMP_ABSOLUTE	18
211	>>	82 LOAD_GLOBAL	2 (_coconut)
		84 LOAD_METHOD	3 (ValueError)
		86 LOAD_CONST	4 ('invalid arguments to ')
		88 LOAD_GLOBAL	2 (_coconut)
		90 LOAD_METHOD	4 (repr)
		92 LOAD_FAST	0 (self)
		94 CALL_METHOD	1
		96 BINARY_ADD	
		98 CALL_METHOD	1
		100 RAISE_VARARGS	1
		102 JUMP_ABSOLUTE	18
212	>>	104 LOAD_FAST	3 (arg)
		106 RETURN_VALUE	

Disassembly of <code object __repr__ at 0x7f6cf329c2f0, file "<dis>", line 213>:

214		0 LOAD_GLOBAL	0 (_coconut)
		2 LOAD_METHOD	1 (repr)
		4 LOAD_FAST	0 (self)

6	LOAD_ATTR	2 (func)
8	CALL_METHOD	1
10	LOAD_CONST	1 (' ')
12	BINARY_ADD	
14	LOAD_CONST	1 (' ')
16	LOAD_METHOD	3 (join)
18	LOAD_CONST	2 (<code object <genexpr> at 0x7f6cf329c190, file "<dis>", line 214>)
20	LOAD_CONST	3 ('_coconut_base_compose.__repr__.<local>')
22	MAKE_FUNCTION	0
24	LOAD_FAST	0 (self)
26	LOAD_ATTR	4 (funcstars)
28	GET_ITER	
30	CALL_FUNCTION	1
32	CALL_METHOD	1
34	BINARY_ADD	
36	RETURN_VALUE	

Disassembly of <code object <genexpr> at 0x7f6cf329c190, file "<dis>", line 214>:

214	0	LOAD_FAST	0 (.0)
>>	2	FOR_ITER	48 (to 52)
	4	UNPACK_SEQUENCE	2
	6	STORE_FAST	1 (f)
	8	STORE_FAST	2 (star)
	10	LOAD_FAST	2 (star)
	12	LOAD_CONST	0 (1)
	14	COMPARE_OP	2 (==)
	16	POP_JUMP_IF_FALSE	22
	18	LOAD_CONST	1 ('..*> ')
	20	JUMP_FORWARD	14 (to 36)
>>	22	LOAD_FAST	2 (star)
	24	LOAD_CONST	2 (2)
	26	COMPARE_OP	2 (==)
	28	POP_JUMP_IF_FALSE	34
	30	LOAD_CONST	3 ('..**>')
	32	JUMP_FORWARD	2 (to 36)
>>	34	LOAD_CONST	4 ('..> ')
>>	36	LOAD_GLOBAL	0 (_coconut)
	38	LOAD_METHOD	1 (repr)

```

40 LOAD_FAST          1 (f)
42 CALL_METHOD        1
44 BINARY_ADD
46 YIELD_VALUE
48 POP_TOP
50 JUMP_ABSOLUTE      2
>> 52 LOAD_CONST      5 (None)
54 RETURN_VALUE

```

Disassembly of <code object __reduce__ at 0x7f6cf329c3a0, file "<dis>", line 215>:

```

216      0 LOAD_FAST          0 (self)
      2 LOAD_ATTR            0 (__class__)
      4 LOAD_FAST          0 (self)
      6 LOAD_ATTR            1 (func)
      8 BUILD_TUPLE        1
     10 LOAD_GLOBAL          2 (_coconut)
     12 LOAD_METHOD          3 (tuple)
     14 LOAD_FAST          0 (self)
     16 LOAD_ATTR            4 (funcstars)
     18 CALL_METHOD          1
     20 BINARY_ADD
     22 BUILD_TUPLE        2
     24 RETURN_VALUE

```

Disassembly of <code object __get__ at 0x7f6cf329c450, file "<dis>", line 217>:

```

218      0 LOAD_GLOBAL          0 (_coconut)
      2 LOAD_ATTR            1 (functools)
      4 LOAD_METHOD          2 (partial)
      6 LOAD_FAST          0 (self)
      8 LOAD_FAST          1 (obj)
     10 CALL_METHOD          2
     12 RETURN_VALUE

```

Disassembly of <code object _coconut_forward_compose at 0x7f6cf329c660, file "<dis>":

```

219      0 LOAD_GLOBAL          0 (_coconut_base_compose)
      2 LOAD_FAST          0 (func)
      4 BUILD_TUPLE        1
      6 LOAD_CONST          1 (<code object <genexpr> at 0x7f6cf329c5

```

```

      8 LOAD_CONST          2 ('_coconut_forward_compose.<locals>.<ge
10 MAKE_FUNCTION          0
12 LOAD_FAST             1 (funcs)
14 GET_ITER
16 CALL_FUNCTION          1
18 BUILD_TUPLE_UNPACK_WITH_CALL    2
20 CALL_FUNCTION_EX       0
22 RETURN_VALUE

```

Disassembly of <code object <genexpr> at 0x7f6cf329c5b0, file "<dis>", line 219>:

```

219      0 LOAD_FAST          0 (.0)
    >>  2 FOR_ITER             14 (to 18)
      4 STORE_FAST           1 (f)
      6 LOAD_FAST           1 (f)
      8 LOAD_CONST           0 (0)
     10 BUILD_TUPLE           2
     12 YIELD_VALUE
     14 POP_TOP
     16 JUMP_ABSOLUTE         2
    >>  18 LOAD_CONST           1 (None)
     20 RETURN_VALUE

```

Disassembly of <code object _coconut_back_compose at 0x7f6cf329c710, file "<dis>",

```

220      0 LOAD_GLOBAL          0 (_coconut_forward_compose)
      2 LOAD_GLOBAL          1 (_coconut)
      4 LOAD_METHOD           2 (reversed)
      6 LOAD_FAST             0 (funcs)
      8 CALL_METHOD           1
     10 CALL_FUNCTION_EX       0
     12 RETURN_VALUE

```

Disassembly of <code object _coconut_forward_star_compose at 0x7f6cf329c870, file

```

221      0 LOAD_GLOBAL          0 (_coconut_base_compose)
      2 LOAD_FAST             0 (func)
      4 BUILD_TUPLE           1
      6 LOAD_CONST            1 (<code object <genexpr> at 0x7f6cf329c7
      8 LOAD_CONST            2 ('_coconut_forward_star_compose.<locals
     10 MAKE_FUNCTION          0

```

```

12 LOAD_FAST          1 (funcs)
14 GET_ITER
16 CALL_FUNCTION      1
18 BUILD_TUPLE_UNPACK_WITH_CALL    2
20 CALL_FUNCTION_EX    0
22 RETURN_VALUE

```

Disassembly of <code object <genexpr> at 0x7f6cf329c7c0, file "<dis>", line 221>:

```

221      0 LOAD_FAST          0 (.0)
      >>  2 FOR_ITER            14 (to 18)
          4 STORE_FAST         1 (f)
          6 LOAD_FAST        1 (f)
          8 LOAD_CONST        0 (1)
         10 BUILD_TUPLE        2
         12 YIELD_VALUE
         14 POP_TOP
         16 JUMP_ABSOLUTE      2
      >>  18 LOAD_CONST        1 (None)
         20 RETURN_VALUE

```

Disassembly of <code object _coconut_back_star_compose at 0x7f6cf329c920, file "<dis>", line 222>:

```

222      0 LOAD_GLOBAL        0 (_coconut_forward_star_compose)
          2 LOAD_GLOBAL        1 (_coconut)
          4 LOAD_METHOD         2 (reversed)
          6 LOAD_FAST          0 (funcs)
          8 CALL_METHOD      1
         10 CALL_FUNCTION_EX    0
         12 RETURN_VALUE

```

Disassembly of <code object _coconut_forward_dubstar_compose at 0x7f6cf329ca80, file "<dis>", line 223>:

```

223      0 LOAD_GLOBAL        0 (_coconut_base_compose)
          2 LOAD_FAST          0 (func)
          4 BUILD_TUPLE        1
          6 LOAD_CONST        1 (<code object <genexpr> at 0x7f6cf329c7c0, file "<dis>", line 221>)
          8 LOAD_CONST        2 ('_coconut_forward_dubstar_compose.<locals>')
         10 MAKE_FUNCTION      0
         12 LOAD_FAST          1 (funcs)
         14 GET_ITER

```

```

16 CALL_FUNCTION          1
18 BUILD_TUPLE_UNPACK_WITH_CALL    2
20 CALL_FUNCTION_EX       0
22 RETURN_VALUE

```

Disassembly of <code object <genexpr> at 0x7f6cf329c9d0, file "<dis>", line 223>:

```

223      0 LOAD_FAST          0 (.0)
      >>  2 FOR_ITER              14 (to 18)
          4 STORE_FAST          1 (f)
          6 LOAD_FAST          1 (f)
          8 LOAD_CONST          0 (2)
         10 BUILD_TUPLE          2
         12 YIELD_VALUE
         14 POP_TOP
         16 JUMP_ABSOLUTE       2
      >>  18 LOAD_CONST          1 (None)
         20 RETURN_VALUE

```

Disassembly of <code object _coconut_back_dubstar_compose at 0x7f6cf329cb30, file

```

224      0 LOAD_GLOBAL          0 (_coconut_forward_dubstar_compose)
          2 LOAD_GLOBAL          1 (_coconut)
          4 LOAD_METHOD          2 (reversed)
          6 LOAD_FAST          0 (funcs)
          8 CALL_METHOD          1
         10 CALL_FUNCTION_EX       0
         12 RETURN_VALUE

```

Disassembly of <code object _coconut_pipe at 0x7f6cf329cbe0, file "<dis>", line 22

```

225      0 LOAD_FAST          1 (f)
          2 LOAD_FAST          0 (x)
          4 CALL_FUNCTION          1
          6 RETURN_VALUE

```

Disassembly of <code object _coconut_star_pipe at 0x7f6cf329cc90, file "<dis>", li

```

226      0 LOAD_FAST          1 (f)
          2 LOAD_FAST          0 (xs)
          4 CALL_FUNCTION_EX       0
          6 RETURN_VALUE

```

```

Disassembly of <code object _coconut_dubstar_pipe at 0x7f6cf329cd40, file "<dis>",
227      0 LOAD_FAST          1 (f)
        2 BUILD_TUPLE          0
        4 LOAD_FAST          0 (kws)
        6 CALL_FUNCTION_EX      1
        8 RETURN_VALUE

```

```

Disassembly of <code object _coconut_back_pipe at 0x7f6cf329cdf0, file "<dis>", li
228      0 LOAD_FAST          0 (f)
        2 LOAD_FAST          1 (x)
        4 CALL_FUNCTION        1
        6 RETURN_VALUE

```

```

Disassembly of <code object _coconut_back_star_pipe at 0x7f6cf329cea0, file "<dis>
229      0 LOAD_FAST          0 (f)
        2 LOAD_FAST          1 (xs)
        4 CALL_FUNCTION_EX      0
        6 RETURN_VALUE

```

```

Disassembly of <code object _coconut_back_dubstar_pipe at 0x7f6cf329cf50, file "<d
230      0 LOAD_FAST          0 (f)
        2 BUILD_TUPLE          0
        4 LOAD_FAST          1 (kws)
        6 CALL_FUNCTION_EX      1
        8 RETURN_VALUE

```

```

Disassembly of <code object _coconut_assert at 0x7f6cf329e030, file "<dis>", line
231      0 LOAD_FAST          0 (cond)
        2 POP_JUMP_IF_TRUE      34
        4 LOAD_GLOBAL          0 (AssertionError)
        6 LOAD_FAST          1 (msg)
        8 LOAD_CONST          0 (None)
       10 COMPARE_OP            9 (is not)
       12 POP_JUMP_IF_FALSE    18
       14 LOAD_FAST          1 (msg)
       16 JUMP_FORWARD          12 (to 30)
>>    18 LOAD_CONST          1 (('assert) got falsey value ')

```

```

20 LOAD_GLOBAL          1 (_coconut)
22 LOAD_METHOD          2 (repr)
24 LOAD_FAST            0 (cond)
26 CALL_METHOD          1
28 BINARY_ADD
>> 30 CALL_FUNCTION      1
32 RAISE_VARARGS        1
>> 34 LOAD_CONST        0 (None)
36 RETURN_VALUE

```

Disassembly of <code object _coconut_bool_and at 0x7f6cf329e0e0, file "<dis>", line

```

232      0 LOAD_FAST      0 (a)
      2 JUMP_IF_FALSE_OR_POP 6
      4 LOAD_FAST      1 (b)
>>     6 RETURN_VALUE

```

Disassembly of <code object _coconut_bool_or at 0x7f6cf329e190, file "<dis>", line

```

233      0 LOAD_FAST      0 (a)
      2 JUMP_IF_TRUE_OR_POP 6
      4 LOAD_FAST      1 (b)
>>     6 RETURN_VALUE

```

Disassembly of <code object _coconut_none_coalesce at 0x7f6cf329e240, file "<dis>"

```

234      0 LOAD_FAST      0 (a)
      2 LOAD_CONST      0 (None)
      4 COMPARE_OP      9 (is not)
      6 POP_JUMP_IF_FALSE 12
      8 LOAD_FAST      0 (a)
     10 RETURN_VALUE
>>     12 LOAD_FAST      1 (b)
     14 RETURN_VALUE

```

Disassembly of <code object _coconut_minus at 0x7f6cf329e2f0, file "<dis>", line 2

```

236      0 LOAD_FAST      1 (rest)
      2 POP_JUMP_IF_TRUE 10

237      4 LOAD_FAST      0 (a)
      6 UNARY_NEGATIVE

```



```

      8 RETURN_VALUE

238    >> 10 LOAD_FAST          1 (rest)
        12 GET_ITER
        >> 14 FOR_ITER          12 (to 28)
        16 STORE_FAST         2 (b)

239        18 LOAD_FAST         0 (a)
        20 LOAD_FAST         2 (b)
        22 BINARY_SUBTRACT
        24 STORE_FAST         0 (a)
        26 JUMP_ABSOLUTE      14

240    >> 28 LOAD_FAST          0 (a)
        30 RETURN_VALUE

```

Disassembly of <code object tee at 0x7f6cf329e5b0, file "<dis>", line 241>:

```

243        0 LOAD_FAST          1 (n)
        2 LOAD_CONST          1 (0)
        4 COMPARE_OP          5 (>=)
        6 POP_JUMP_IF_FALSE    38
        8 LOAD_GLOBAL          0 (_coconut)
       10 LOAD_METHOD            1 (isinstance)
       12 LOAD_DEREF            0 (iterable)
       14 LOAD_GLOBAL          0 (_coconut)
       16 LOAD_ATTR              2 (tuple)
       18 LOAD_GLOBAL          0 (_coconut)
       20 LOAD_ATTR              3 (frozenset)
       22 BUILD_TUPLE           2
       24 CALL_METHOD            2
       26 POP_JUMP_IF_FALSE    38

244        28 LOAD_DEREF            0 (iterable)
       30 BUILD_TUPLE           1
       32 LOAD_FAST             1 (n)
       34 BINARY_MULTIPLY
       36 RETURN_VALUE

```

```

245    >> 38 LOAD_FAST          1 (n)
          40 LOAD_CONST          1 (0)
          42 COMPARE_OP            4 (>)
          44 POP_JUMP_IF_FALSE    114
          46 LOAD_GLOBAL          0 (_coconut)
          48 LOAD_METHOD          4 (hasattr)
          50 LOAD_DEREF           0 (iterable)
          52 LOAD_CONST          2 ('__copy__')
          54 CALL_METHOD           2
          56 POP_JUMP_IF_TRUE     74
          58 LOAD_GLOBAL          0 (_coconut)
          60 LOAD_METHOD          1 (isinstance)
          62 LOAD_DEREF           0 (iterable)
          64 LOAD_GLOBAL          0 (_coconut)
          66 LOAD_ATTR            5 (abc)
          68 LOAD_ATTR            6 (Sequence)
          70 CALL_METHOD           2
          72 POP_JUMP_IF_FALSE    114

246    >> 74 LOAD_DEREF           0 (iterable)
          76 BUILD_TUPLE           1
          78 LOAD_GLOBAL          0 (_coconut)
          80 LOAD_METHOD          2 (tuple)
          82 LOAD_CLOSURE          0 (iterable)
          84 BUILD_TUPLE           1
          86 LOAD_CONST          3 (<code object <genexpr> at 0x7f6cf329e3
          88 LOAD_CONST          4 ('tee.<locals>.<genexpr>')
          90 MAKE_FUNCTION          8 (closure)
          92 LOAD_GLOBAL          0 (_coconut)
          94 LOAD_METHOD          7 (range)
          96 LOAD_FAST            1 (n)
          98 LOAD_CONST          5 (1)
         100 BINARY_SUBTRACT
         102 CALL_METHOD           1
         104 GET_ITER
         106 CALL_FUNCTION          1
         108 CALL_METHOD           1
         110 BINARY_ADD

```

```

112 RETURN_VALUE

247    >> 114 LOAD_GLOBAL          0 (_coconut)
        116 LOAD_ATTR             8 (itertools)
        118 LOAD_METHOD           9 (tee)
        120 LOAD_DEREF            0 (iterable)
        122 LOAD_FAST             1 (n)
        124 CALL_METHOD           2
        126 RETURN_VALUE

```

Disassembly of <code object <genexpr> at 0x7f6cf329e3a0, file "<dis>", line 246>:

```

246      0 LOAD_FAST              0 (.0)
    >>   2 FOR_ITER                18 (to 22)
        4 STORE_FAST              1 (_)
        6 LOAD_GLOBAL            0 (_coconut)
        8 LOAD_ATTR               1 (copy)
       10 LOAD_METHOD              1 (copy)
       12 LOAD_DEREF              0 (iterable)
       14 CALL_METHOD            1
       16 YIELD_VALUE
       18 POP_TOP
       20 JUMP_ABSOLUTE          2
    >>   22 LOAD_CONST              0 (None)
       24 RETURN_VALUE

```

Disassembly of <code object reiterable at 0x7f6cf329ed40, file "<dis>", line 248>:

```

248      0 LOAD_NAME                0 (__name__)
        2 STORE_NAME               1 (__module__)
        4 LOAD_CONST               0 ('reiterable')
        6 STORE_NAME              2 (__qualname__)

249      8 LOAD_CONST                 1 ('Allows an iterator to be iterated over')
       10 STORE_NAME                 3 (__doc__)

250     12 LOAD_CONST                 2 (('iter',))
       14 STORE_NAME                 4 (__slots__)

251     16 LOAD_CONST                 3 (<code object __init__ at 0x7f6cf329e450>)

```

	18 LOAD_CONST	4 ('reiterable.__init__')
	20 MAKE_FUNCTION	0
	22 STORE_NAME	5 (__init__)
253	24 LOAD_CONST	5 (<code object _get_new_iter at 0x7f6cf329e710>)
	26 LOAD_CONST	6 ('reiterable._get_new_iter')
	28 MAKE_FUNCTION	0
	30 STORE_NAME	6 (_get_new_iter)
256	32 LOAD_CONST	7 (<code object __iter__ at 0x7f6cf329e710>)
	34 LOAD_CONST	8 ('reiterable.__iter__')
	36 MAKE_FUNCTION	0
	38 STORE_NAME	7 (__iter__)
258	40 LOAD_CONST	9 (<code object __getitem__ at 0x7f6cf329e710>)
	42 LOAD_CONST	10 ('reiterable.__getitem__')
	44 MAKE_FUNCTION	0
	46 STORE_NAME	8 (__getitem__)
260	48 LOAD_CONST	11 (<code object __reversed__ at 0x7f6cf329e710>)
	50 LOAD_CONST	12 ('reiterable.__reversed__')
	52 MAKE_FUNCTION	0
	54 STORE_NAME	9 (__reversed__)
262	56 LOAD_CONST	13 (<code object __len__ at 0x7f6cf329e710>)
	58 LOAD_CONST	14 ('reiterable.__len__')
	60 MAKE_FUNCTION	0
	62 STORE_NAME	10 (__len__)
264	64 LOAD_CONST	15 (<code object __repr__ at 0x7f6cf329e710>)
	66 LOAD_CONST	16 ('reiterable.__repr__')
	68 MAKE_FUNCTION	0
	70 STORE_NAME	11 (__repr__)
266	72 LOAD_CONST	17 (<code object __reduce__ at 0x7f6cf329e710>)
	74 LOAD_CONST	18 ('reiterable.__reduce__')
	76 MAKE_FUNCTION	0
	78 STORE_NAME	12 (__reduce__)

268	80 LOAD_CONST	19 (<code object __copy__ at 0x7f6cf329eb3
	82 LOAD_CONST	20 ('reiterable.__copy__')
	84 MAKE_FUNCTION	0
	86 STORE_NAME	13 (__copy__)
270	88 LOAD_CONST	21 (<code object __fmap__ at 0x7f6cf329ebe
	90 LOAD_CONST	22 ('reiterable.__fmap__')
	92 MAKE_FUNCTION	0
	94 STORE_NAME	14 (__fmap__)
	96 LOAD_CONST	23 (None)
	98 RETURN_VALUE	

Disassembly of <code object __init__ at 0x7f6cf329e450, file "<dis>", line 251>:

252	0 LOAD_FAST	1 (iterable)
	2 LOAD_FAST	0 (self)
	4 STORE_ATTR	0 (iter)
	6 LOAD_CONST	0 (None)
	8 RETURN_VALUE	

Disassembly of <code object _get_new_iter at 0x7f6cf329e660, file "<dis>", line 253>:

254	0 LOAD_GLOBAL	0 (_coconut_tee)
	2 LOAD_FAST	0 (self)
	4 LOAD_ATTR	1 (iter)
	6 CALL_FUNCTION	1
	8 UNPACK_SEQUENCE	2
	10 LOAD_FAST	0 (self)
	12 STORE_ATTR	1 (iter)
	14 STORE_FAST	1 (new_iter)
255	16 LOAD_FAST	1 (new_iter)
	18 RETURN_VALUE	

Disassembly of <code object __iter__ at 0x7f6cf329e710, file "<dis>", line 256>:

257	0 LOAD_GLOBAL	0 (_coconut)
	2 LOAD_METHOD	1 (iter)
	4 LOAD_FAST	0 (self)
	6 LOAD_METHOD	2 (_get_new_iter)

	8	CALL_METHOD	0
	10	CALL_METHOD	1
	12	RETURN_VALUE	

Disassembly of <code object __getitem__ at 0x7f6cf329e7c0, file "<dis>", line 258>

259	0	LOAD_GLOBAL	0 (_coconut_igetitem)
	2	LOAD_FAST	0 (self)
	4	LOAD_METHOD	1 (_get_new_iter)
	6	CALL_METHOD	0
	8	LOAD_FAST	1 (index)
	10	CALL_FUNCTION	2
	12	RETURN_VALUE	

Disassembly of <code object __reversed__ at 0x7f6cf329e870, file "<dis>", line 260>

261	0	LOAD_GLOBAL	0 (_coconut_reversed)
	2	LOAD_FAST	0 (self)
	4	LOAD_METHOD	1 (_get_new_iter)
	6	CALL_METHOD	0
	8	CALL_FUNCTION	1
	10	RETURN_VALUE	

Disassembly of <code object __len__ at 0x7f6cf329e920, file "<dis>", line 262>:

263	0	LOAD_GLOBAL	0 (_coconut)
	2	LOAD_METHOD	1 (len)
	4	LOAD_FAST	0 (self)
	6	LOAD_ATTR	2 (iter)
	8	CALL_METHOD	1
	10	RETURN_VALUE	

Disassembly of <code object __repr__ at 0x7f6cf329e9d0, file "<dis>", line 264>:

265	0	LOAD_CONST	1 ('reiterable(%r)')
	2	LOAD_FAST	0 (self)
	4	LOAD_ATTR	0 (iter)
	6	BUILD_TUPLE	1
	8	BINARY_MODULO	
	10	RETURN_VALUE	

Disassembly of <code object __reduce__ at 0x7f6cf329ea80, file "<dis>", line 266>:

267	0	LOAD_FAST	0	(self)
	2	LOAD_ATTR	0	(__class__)
	4	LOAD_FAST	0	(self)
	6	LOAD_ATTR	1	(iter)
	8	BUILD_TUPLE	1	
	10	BUILD_TUPLE	2	
	12	RETURN_VALUE		

Disassembly of <code object __copy__ at 0x7f6cf329eb30, file "<dis>", line 268>:

269	0	LOAD_FAST	0	(self)
	2	LOAD_METHOD	0	(__class__)
	4	LOAD_FAST	0	(self)
	6	LOAD_METHOD	1	(_get_new_iter)
	8	CALL_METHOD	0	
	10	CALL_METHOD	1	
	12	RETURN_VALUE		

Disassembly of <code object __fmap__ at 0x7f6cf329ebe0, file "<dis>", line 270>:

271	0	LOAD_GLOBAL	0	(_coconut_map)
	2	LOAD_FAST	1	(func)
	4	LOAD_FAST	0	(self)
	6	CALL_FUNCTION	2	
	8	RETURN_VALUE		

Disassembly of <code object scan at 0x7f6cf32a3240, file "<dis>", line 272>:

272	0	LOAD_NAME	0	(__name__)
	2	STORE_NAME	1	(__module__)
	4	LOAD_CONST	0	('scan')
	6	STORE_NAME	2	(__qualname__)
273	8	LOAD_CONST	1	('Reduce func over iterable, yielding i
	10	STORE_NAME	3	(__doc__)
275	12	LOAD_CONST	2	((('func', 'iter', 'initializer'))
	14	STORE_NAME	4	(__slots__)
276	16	LOAD_NAME	5	(_coconut_sentinel)
	18	BUILD_TUPLE	1	

	20 LOAD_CONST	3 (<code object __init__ at 0x7f6cf329ec90)
	22 LOAD_CONST	4 ('scan.__init__')
	24 MAKE_FUNCTION	1 (defaults)
	26 STORE_NAME	6 (__init__)
280	28 LOAD_CONST	5 (<code object __iter__ at 0x7f6cf329edf0)
	30 LOAD_CONST	6 ('scan.__iter__')
	32 MAKE_FUNCTION	0
	34 STORE_NAME	7 (__iter__)
290	36 LOAD_CONST	7 (<code object __len__ at 0x7f6cf329eea0)
	38 LOAD_CONST	8 ('scan.__len__')
	40 MAKE_FUNCTION	0
	42 STORE_NAME	8 (__len__)
292	44 LOAD_CONST	9 (<code object __repr__ at 0x7f6cf329ef50)
	46 LOAD_CONST	10 ('scan.__repr__')
	48 MAKE_FUNCTION	0
	50 STORE_NAME	9 (__repr__)
294	52 LOAD_CONST	11 (<code object __reduce__ at 0x7f6cf32a3000)
	54 LOAD_CONST	12 ('scan.__reduce__')
	56 MAKE_FUNCTION	0
	58 STORE_NAME	10 (__reduce__)
296	60 LOAD_CONST	13 (<code object __copy__ at 0x7f6cf32a30e0)
	62 LOAD_CONST	14 ('scan.__copy__')
	64 MAKE_FUNCTION	0
	66 STORE_NAME	11 (__copy__)
298	68 LOAD_CONST	15 (<code object __fmap__ at 0x7f6cf32a3190)
	70 LOAD_CONST	16 ('scan.__fmap__')
	72 MAKE_FUNCTION	0
	74 STORE_NAME	12 (__fmap__)
	76 LOAD_CONST	17 (None)
	78 RETURN_VALUE	

Disassembly of <code object __init__ at 0x7f6cf329ec90, file "<dis>", line 276>:

277	0 LOAD_FAST	1 (function)
	2 LOAD_FAST	0 (self)
	4 STORE_ATTR	0 (func)
278	6 LOAD_FAST	2 (iterable)
	8 LOAD_FAST	0 (self)
	10 STORE_ATTR	1 (iter)
279	12 LOAD_FAST	3 (initializer)
	14 LOAD_FAST	0 (self)
	16 STORE_ATTR	2 (initializer)
	18 LOAD_CONST	0 (None)
	20 RETURN_VALUE	

Disassembly of <code object __iter__ at 0x7f6cf329edf0, file "<dis>", line 280>:

281	0 LOAD_FAST	0 (self)
	2 LOAD_ATTR	0 (initializer)
	4 STORE_FAST	1 (acc)
282	6 LOAD_FAST	1 (acc)
	8 LOAD_GLOBAL	1 (_coconut_sentinel)
	10 COMPARE_OP	9 (is not)
	12 POP_JUMP_IF_FALSE	20
283	14 LOAD_FAST	1 (acc)
	16 YIELD_VALUE	
	18 POP_TOP	
284	>> 20 LOAD_FAST	0 (self)
	22 LOAD_ATTR	2 (iter)
	24 GET_ITER	
	>> 26 FOR_ITER	36 (to 64)
	28 STORE_FAST	2 (item)
285	30 LOAD_FAST	1 (acc)
	32 LOAD_GLOBAL	1 (_coconut_sentinel)
	34 COMPARE_OP	8 (is)
	36 POP_JUMP_IF_FALSE	44

286		38 LOAD_FAST	2 (item)
		40 STORE_FAST	1 (acc)
		42 JUMP_FORWARD	12 (to 56)
288	>>	44 LOAD_FAST	0 (self)
		46 LOAD_METHOD	3 (func)
		48 LOAD_FAST	1 (acc)
		50 LOAD_FAST	2 (item)
		52 CALL_METHOD	2
		54 STORE_FAST	1 (acc)
289	>>	56 LOAD_FAST	1 (acc)
		58 YIELD_VALUE	
		60 POP_TOP	
		62 JUMP_ABSOLUTE	26
	>>	64 LOAD_CONST	0 (None)
		66 RETURN_VALUE	

Disassembly of <code object __len__ at 0x7f6cf329eea0, file "<dis>", line 290>:

291		0 LOAD_GLOBAL	0 (_coconut)
		2 LOAD_METHOD	1 (len)
		4 LOAD_FAST	0 (self)
		6 LOAD_ATTR	2 (iter)
		8 CALL_METHOD	1
		10 RETURN_VALUE	

Disassembly of <code object __repr__ at 0x7f6cf329ef50, file "<dis>", line 292>:

293		0 LOAD_CONST	1 ('scan(%r, %r)')
		2 LOAD_FAST	0 (self)
		4 LOAD_ATTR	0 (func)
		6 LOAD_FAST	0 (self)
		8 LOAD_ATTR	1 (iter)
		10 BUILD_TUPLE	2
		12 BINARY_MODULO	
		14 RETURN_VALUE	

Disassembly of <code object __reduce__ at 0x7f6cf32a3030, file "<dis>", line 294>:

295	0	LOAD_FAST	0	(self)
	2	LOAD_ATTR	0	(__class__)
	4	LOAD_FAST	0	(self)
	6	LOAD_ATTR	1	(func)
	8	LOAD_FAST	0	(self)
	10	LOAD_ATTR	2	(iter)
	12	BUILD_TUPLE	2	
	14	BUILD_TUPLE	2	
	16	RETURN_VALUE		

Disassembly of <code object __copy__ at 0x7f6cf32a30e0, file "<dis>", line 296>:

297	0	LOAD_FAST	0	(self)
	2	LOAD_METHOD	0	(__class__)
	4	LOAD_FAST	0	(self)
	6	LOAD_ATTR	1	(func)
	8	LOAD_GLOBAL	2	(_coconut)
	10	LOAD_ATTR	3	(copy)
	12	LOAD_METHOD	3	(copy)
	14	LOAD_FAST	0	(self)
	16	LOAD_ATTR	4	(iter)
	18	CALL_METHOD	1	
	20	CALL_METHOD	2	
	22	RETURN_VALUE		

Disassembly of <code object __fmap__ at 0x7f6cf32a3190, file "<dis>", line 298>:

299	0	LOAD_GLOBAL	0	(_coconut_map)
	2	LOAD_FAST	1	(func)
	4	LOAD_FAST	0	(self)
	6	CALL_FUNCTION	2	
	8	RETURN_VALUE		

Disassembly of <code object reversed at 0x7f6cf32a3d40, file "<dis>", line 300>:

300	0	LOAD_NAME	0	(__name__)
	2	STORE_NAME	1	(__module__)
	4	LOAD_CONST	0	('reversed')
	6	STORE_NAME	2	(__qualname__)
301	8	LOAD_CONST	1	((('iter',))

		10 STORE_NAME	3 (__slots__)
302		12 LOAD_NAME	4 (hasattr)
		14 LOAD_NAME	5 (_coconut)
		16 LOAD_ATTR	6 (map)
		18 LOAD_CONST	2 ('__doc__')
		20 CALL_FUNCTION	2
		22 POP_JUMP_IF_FALSE	32
303		24 LOAD_NAME	5 (_coconut)
		26 LOAD_ATTR	7 (reversed)
		28 LOAD_ATTR	8 (__doc__)
		30 STORE_NAME	8 (__doc__)
304	>>	32 LOAD_CONST	3 (<code object __new__ at 0x7f6cf32a32f0)
		34 LOAD_CONST	4 ('reversed.__new__')
		36 MAKE_FUNCTION	0
		38 STORE_NAME	9 (__new__)
310		40 LOAD_CONST	5 (<code object __init__ at 0x7f6cf32a33a0)
		42 LOAD_CONST	6 ('reversed.__init__')
		44 MAKE_FUNCTION	0
		46 STORE_NAME	10 (__init__)
312		48 LOAD_CONST	7 (<code object __iter__ at 0x7f6cf32a3450)
		50 LOAD_CONST	8 ('reversed.__iter__')
		52 MAKE_FUNCTION	0
		54 STORE_NAME	11 (__iter__)
314		56 LOAD_CONST	9 (<code object __getitem__ at 0x7f6cf32a3500)
		58 LOAD_CONST	10 ('reversed.__getitem__')
		60 MAKE_FUNCTION	0
		62 STORE_NAME	12 (__getitem__)
318		64 LOAD_CONST	11 (<code object __reversed__ at 0x7f6cf32a35b0)
		66 LOAD_CONST	12 ('reversed.__reversed__')
		68 MAKE_FUNCTION	0
		70 STORE_NAME	13 (__reversed__)

320	72 LOAD_CONST	13 (<code object __len__ at 0x7f6cf32a3660
	74 LOAD_CONST	14 ('reversed.__len__')
	76 MAKE_FUNCTION	0
	78 STORE_NAME	14 (__len__)
322	80 LOAD_CONST	15 (<code object __repr__ at 0x7f6cf32a3710
	82 LOAD_CONST	16 ('reversed.__repr__')
	84 MAKE_FUNCTION	0
	86 STORE_NAME	15 (__repr__)
324	88 LOAD_CONST	17 (<code object __hash__ at 0x7f6cf32a37c0
	90 LOAD_CONST	18 ('reversed.__hash__')
	92 MAKE_FUNCTION	0
	94 STORE_NAME	16 (__hash__)
326	96 LOAD_CONST	19 (<code object __reduce__ at 0x7f6cf32a3800
	98 LOAD_CONST	20 ('reversed.__reduce__')
	100 MAKE_FUNCTION	0
	102 STORE_NAME	17 (__reduce__)
328	104 LOAD_CONST	21 (<code object __copy__ at 0x7f6cf32a3920
	106 LOAD_CONST	22 ('reversed.__copy__')
	108 MAKE_FUNCTION	0
	110 STORE_NAME	18 (__copy__)
330	112 LOAD_CONST	23 (<code object __eq__ at 0x7f6cf32a39d0
	114 LOAD_CONST	24 ('reversed.__eq__')
	116 MAKE_FUNCTION	0
	118 STORE_NAME	19 (__eq__)
332	120 LOAD_CONST	25 (<code object __contains__ at 0x7f6cf32a3a80
	122 LOAD_CONST	26 ('reversed.__contains__')
	124 MAKE_FUNCTION	0
	126 STORE_NAME	20 (__contains__)
334	128 LOAD_CONST	27 (<code object count at 0x7f6cf32a3b30
	130 LOAD_CONST	28 ('reversed.count')

	132	MAKE_FUNCTION	0
	134	STORE_NAME	21 (count)
337	136	LOAD_CONST	29 (<code object index at 0x7f6cf32a3be0,
	138	LOAD_CONST	30 ('reversed.index')
	140	MAKE_FUNCTION	0
	142	STORE_NAME	22 (index)
340	144	LOAD_CONST	31 (<code object __fmap__ at 0x7f6cf32a3c9
	146	LOAD_CONST	32 ('reversed.__fmap__')
	148	MAKE_FUNCTION	0
	150	STORE_NAME	23 (__fmap__)
	152	LOAD_CONST	33 (None)
	154	RETURN_VALUE	

Disassembly of <code object __new__ at 0x7f6cf32a32f0, file "<dis>", line 304>:

305	0	LOAD_GLOBAL	0 (_coconut)
	2	LOAD_METHOD	1 (isinstance)
	4	LOAD_FAST	1 (iterable)
	6	LOAD_GLOBAL	0 (_coconut)
	8	LOAD_ATTR	2 (range)
	10	CALL_METHOD	2
	12	POP_JUMP_IF_FALSE	28
306	14	LOAD_FAST	1 (iterable)
	16	LOAD_CONST	0 (None)
	18	LOAD_CONST	0 (None)
	20	LOAD_CONST	1 (-1)
	22	BUILD_SLICE	3
	24	BINARY_SUBSCR	
	26	RETURN_VALUE	
307	>>	28	LOAD_GLOBAL
		30	LOAD_METHOD
		32	LOAD_FAST
		34	LOAD_CONST
		36	CALL_METHOD
		38	POP_JUMP_IF_FALSE
			0 (_coconut)
			3 (hasattr)
			1 (iterable)
			2 ('__reversed__')
			2
			60

		40	LOAD_GLOBAL	0	(_coconut)
		42	LOAD_METHOD	1	(isinstance)
		44	LOAD_FAST	1	(iterable)
		46	LOAD_GLOBAL	0	(_coconut)
		48	LOAD_ATTR	4	(list)
		50	LOAD_GLOBAL	0	(_coconut)
		52	LOAD_ATTR	5	(tuple)
		54	BUILD_TUPLE	2	
		56	CALL_METHOD	2	
		58	POP_JUMP_IF_FALSE	72	
308	>>	60	LOAD_GLOBAL	0	(_coconut)
		62	LOAD_ATTR	6	(object)
		64	LOAD_METHOD	7	(__new__)
		66	LOAD_FAST	0	(cls)
		68	CALL_METHOD	1	
		70	RETURN_VALUE		
309	>>	72	LOAD_GLOBAL	0	(_coconut)
		74	LOAD_METHOD	8	(reversed)
		76	LOAD_FAST	1	(iterable)
		78	CALL_METHOD	1	
		80	RETURN_VALUE		

Disassembly of <code object __init__ at 0x7f6cf32a33a0, file "<dis>", line 310>:

311		0	LOAD_FAST	1	(iterable)
		2	LOAD_FAST	0	(self)
		4	STORE_ATTR	0	(iter)
		6	LOAD_CONST	0	(None)
		8	RETURN_VALUE		

Disassembly of <code object __iter__ at 0x7f6cf32a3450, file "<dis>", line 312>:

313		0	LOAD_GLOBAL	0	(_coconut)
		2	LOAD_METHOD	1	(iter)
		4	LOAD_GLOBAL	0	(_coconut)
		6	LOAD_METHOD	2	(reversed)
		8	LOAD_FAST	0	(self)
		10	LOAD_ATTR	1	(iter)

12	CALL_METHOD	1
14	CALL_METHOD	1
16	RETURN_VALUE	

Disassembly of <code object __getitem__ at 0x7f6cf32a35b0, file "<dis>", line 314>

315	0	LOAD_GLOBAL	0	(_coconut)
	2	LOAD_METHOD	1	(isinstance)
	4	LOAD_FAST	1	(index)
	6	LOAD_GLOBAL	0	(_coconut)
	8	LOAD_ATTR	2	(slice)
	10	CALL_METHOD	2	
	12	POP_JUMP_IF_FALSE	94	
316	14	LOAD_GLOBAL	3	(_coconut_igetitem)
	16	LOAD_FAST	0	(self)
	18	LOAD_ATTR	4	(iter)
	20	LOAD_GLOBAL	0	(_coconut)
	22	LOAD_METHOD	2	(slice)
	24	LOAD_FAST	1	(index)
	26	LOAD_ATTR	5	(start)
	28	LOAD_CONST	0	(None)
	30	COMPARE_OP	9	(is not)
	32	POP_JUMP_IF_FALSE	46	
	34	LOAD_FAST	1	(index)
	36	LOAD_ATTR	5	(start)
	38	LOAD_CONST	1	(1)
	40	BINARY_ADD		
	42	UNARY_NEGATIVE		
	44	JUMP_FORWARD	2	(to 48)
>>	46	LOAD_CONST	0	(None)
>>	48	LOAD_FAST	1	(index)
	50	LOAD_ATTR	6	(stop)
	52	POP_JUMP_IF_FALSE	66	
	54	LOAD_FAST	1	(index)
	56	LOAD_ATTR	6	(stop)
	58	LOAD_CONST	1	(1)
	60	BINARY_ADD		
	62	UNARY_NEGATIVE		


```

        64 JUMP_FORWARD          2 (to 68)
>>    66 LOAD_CONST             0 (None)
>>    68 LOAD_FAST              1 (index)
        70 LOAD_ATTR            7 (step)
        72 LOAD_CONST             0 (None)
        74 COMPARE_OP           9 (is not)
        76 POP_JUMP_IF_FALSE    84
        78 LOAD_FAST              1 (index)
        80 LOAD_ATTR            7 (step)
        82 JUMP_FORWARD          2 (to 86)
>>    84 LOAD_CONST             1 (1)
>>    86 UNARY_NEGATIVE
        88 CALL_METHOD           3
        90 CALL_FUNCTION         2
        92 RETURN_VALUE

317    >>    94 LOAD_GLOBAL        3 (_coconut_igetitem)
        96 LOAD_FAST             0 (self)
        98 LOAD_ATTR            4 (iter)
       100 LOAD_FAST             1 (index)
       102 LOAD_CONST            1 (1)
       104 BINARY_ADD
       106 UNARY_NEGATIVE
       108 CALL_FUNCTION           2
       110 RETURN_VALUE

```

Disassembly of <code object __reversed__ at 0x7f6cf32a3500, file "<dis>", line 318

```

319        0 LOAD_FAST           0 (self)
        2 LOAD_ATTR            0 (iter)
        4 RETURN_VALUE

```

Disassembly of <code object __len__ at 0x7f6cf32a3660, file "<dis>", line 320>:

```

321        0 LOAD_GLOBAL        0 (_coconut)
        2 LOAD_METHOD           1 (len)
        4 LOAD_FAST             0 (self)
        6 LOAD_ATTR            2 (iter)
        8 CALL_METHOD           1
       10 RETURN_VALUE

```

```

Disassembly of <code object __repr__ at 0x7f6cf32a3710, file "<dis>", line 322>:
323      0 LOAD_CONST          1 ('reversed(%r)')
      2 LOAD_FAST              0 (self)
      4 LOAD_ATTR              0 (iter)
      6 BUILD_TUPLE            1
      8 BINARY_MODULO
     10 RETURN_VALUE

Disassembly of <code object __hash__ at 0x7f6cf32a37c0, file "<dis>", line 324>:
325      0 LOAD_GLOBAL          0 (_coconut)
      2 LOAD_METHOD            1 (hash)
      4 LOAD_FAST              0 (self)
      6 LOAD_ATTR              2 (iter)
      8 CALL_METHOD            1
     10 UNARY_NEGATIVE
     12 RETURN_VALUE

Disassembly of <code object __reduce__ at 0x7f6cf32a3870, file "<dis>", line 326>:
327      0 LOAD_FAST              0 (self)
      2 LOAD_ATTR              0 (__class__)
      4 LOAD_FAST              0 (self)
      6 LOAD_ATTR              1 (iter)
      8 BUILD_TUPLE            1
     10 BUILD_TUPLE            2
     12 RETURN_VALUE

Disassembly of <code object __copy__ at 0x7f6cf32a3920, file "<dis>", line 328>:
329      0 LOAD_FAST              0 (self)
      2 LOAD_METHOD            0 (__class__)
      4 LOAD_GLOBAL            1 (_coconut)
      6 LOAD_ATTR              2 (copy)
      8 LOAD_METHOD            2 (copy)
     10 LOAD_FAST              0 (self)
     12 LOAD_ATTR              3 (iter)
     14 CALL_METHOD            1
     16 CALL_METHOD            1
     18 RETURN_VALUE

```

Disassembly of <code object __eq__ at 0x7f6cf32a39d0, file "<dis>", line 330>:

```
331      0 LOAD_GLOBAL      0 (isinstance)
      2 LOAD_FAST            1 (other)
      4 LOAD_FAST            0 (self)
      6 LOAD_ATTR           1 (__class__)
      8 CALL_FUNCTION       2
     10 JUMP_IF_FALSE_OR_POP 22
     12 LOAD_FAST            0 (self)
     14 LOAD_ATTR           2 (iter)
     16 LOAD_FAST            1 (other)
     18 LOAD_ATTR           2 (iter)
     20 COMPARE_OP          2 (==)
    >> 22 RETURN_VALUE
```

Disassembly of <code object __contains__ at 0x7f6cf32a3a80, file "<dis>", line 332>:

```
333      0 LOAD_FAST          1 (elem)
      2 LOAD_FAST            0 (self)
      4 LOAD_ATTR           0 (iter)
      6 COMPARE_OP          6 (in)
      8 RETURN_VALUE
```

Disassembly of <code object count at 0x7f6cf32a3b30, file "<dis>", line 334>:

```
336      0 LOAD_FAST            0 (self)
      2 LOAD_ATTR           0 (iter)
      4 LOAD_METHOD         1 (count)
      6 LOAD_FAST            1 (elem)
      8 CALL_METHOD          1
     10 RETURN_VALUE
```

Disassembly of <code object index at 0x7f6cf32a3be0, file "<dis>", line 337>:

```
339      0 LOAD_GLOBAL         0 (_coconut)
      2 LOAD_METHOD          1 (len)
      4 LOAD_FAST            0 (self)
      6 LOAD_ATTR           2 (iter)
      8 CALL_METHOD          1
     10 LOAD_FAST            0 (self)
     12 LOAD_ATTR           2 (iter)
```

14	LOAD_METHOD	3 (index)
16	LOAD_FAST	1 (elem)
18	CALL_METHOD	1
20	BINARY_SUBTRACT	
22	LOAD_CONST	1 (1)
24	BINARY_SUBTRACT	
26	RETURN_VALUE	

Disassembly of <code object __fmap__ at 0x7f6cf32a3c90, file "<dis>", line 340>:

341	0	LOAD_FAST	0 (self)
	2	LOAD_METHOD	0 (__class__)
	4	LOAD_GLOBAL	1 (_coconut_map)
	6	LOAD_FAST	1 (func)
	8	LOAD_FAST	0 (self)
	10	LOAD_ATTR	2 (iter)
	12	CALL_FUNCTION	2
	14	CALL_METHOD	1
	16	RETURN_VALUE	

Disassembly of <code object map at 0x7f6cf32a6920, file "<dis>", line 342>:

342	0	LOAD_NAME	0 (__name__)
	2	STORE_NAME	1 (__module__)
	4	LOAD_CONST	0 ('map')
	6	STORE_NAME	2 (__qualname__)
343	8	LOAD_CONST	1 (('func', 'iters'))
	10	STORE_NAME	3 (__slots__)
344	12	LOAD_NAME	4 (hasattr)
	14	LOAD_NAME	5 (_coconut)
	16	LOAD_ATTR	6 (map)
	18	LOAD_CONST	2 ('__doc__')
	20	CALL_FUNCTION	2
	22	POP_JUMP_IF_FALSE	32
345	24	LOAD_NAME	5 (_coconut)
	26	LOAD_ATTR	6 (map)
	28	LOAD_ATTR	7 (__doc__)

	30	STORE_NAME	7 (__doc__)
346	>>	32 LOAD_CONST	3 (<code object __new__ at 0x7f6cf32a3df0)
		34 LOAD_CONST	4 ('map.__new__')
		36 MAKE_FUNCTION	0
		38 STORE_NAME	8 (__new__)
351		40 LOAD_CONST	5 (<code object __getitem__ at 0x7f6cf32a3e00)
		42 LOAD_CONST	6 ('map.__getitem__')
		44 MAKE_FUNCTION	0
		46 STORE_NAME	9 (__getitem__)
355		48 LOAD_CONST	7 (<code object __reversed__ at 0x7f6cf32a3e10)
		50 LOAD_CONST	8 ('map.__reversed__')
		52 MAKE_FUNCTION	0
		54 STORE_NAME	10 (__reversed__)
357		56 LOAD_CONST	9 (<code object __len__ at 0x7f6cf32a3e20)
		58 LOAD_CONST	10 ('map.__len__')
		60 MAKE_FUNCTION	0
		62 STORE_NAME	11 (__len__)
359		64 LOAD_CONST	11 (<code object __repr__ at 0x7f6cf32a3e30)
		66 LOAD_CONST	12 ('map.__repr__')
		68 MAKE_FUNCTION	0
		70 STORE_NAME	12 (__repr__)
361		72 LOAD_CONST	13 (<code object __reduce__ at 0x7f6cf32a3e40)
		74 LOAD_CONST	14 ('map.__reduce__')
		76 MAKE_FUNCTION	0
		78 STORE_NAME	13 (__reduce__)
363		80 LOAD_CONST	15 (<code object __reduce_ex__ at 0x7f6cf32a3e50)
		82 LOAD_CONST	16 ('map.__reduce_ex__')
		84 MAKE_FUNCTION	0
		86 STORE_NAME	14 (__reduce_ex__)
365		88 LOAD_CONST	17 (<code object __copy__ at 0x7f6cf32a3e60)

	90	LOAD_CONST	18	('map.__copy__')
	92	MAKE_FUNCTION	0	
	94	STORE_NAME	15	(__copy__)
367	96	LOAD_CONST	19	(<code object __fmap__ at 0x7f6cf32a67c0>)
	98	LOAD_CONST	20	('map.__fmap__')
	100	MAKE_FUNCTION	0	
	102	STORE_NAME	16	(__fmap__)
	104	LOAD_CONST	21	(None)
	106	RETURN_VALUE		

Disassembly of <code object __new__ at 0x7f6cf32a3df0, file "<dis>", line 346>:

347	0	LOAD_GLOBAL	0	(_coconut)
	2	LOAD_ATTR	1	(map)
	4	LOAD_ATTR	2	(__new__)
	6	LOAD_FAST	0	(cls)
	8	LOAD_FAST	1	(function)
	10	BUILD_TUPLE	2	
	12	LOAD_FAST	2	(iterables)
	14	BUILD_TUPLE_UNPACK_WITH_CALL	2	
	16	CALL_FUNCTION_EX	0	
	18	STORE_FAST	3	(new_map)
348	20	LOAD_FAST	1	(function)
	22	LOAD_FAST	3	(new_map)
	24	STORE_ATTR	3	(func)
349	26	LOAD_FAST	2	(iterables)
	28	LOAD_FAST	3	(new_map)
	30	STORE_ATTR	4	(iters)
350	32	LOAD_FAST	3	(new_map)
	34	RETURN_VALUE		

Disassembly of <code object __getitem__ at 0x7f6cf32a60e0, file "<dis>", line 351>:

352	0	LOAD_GLOBAL	0	(_coconut)
	2	LOAD_METHOD	1	(isinstance)
	4	LOAD_DEREF	0	(index)

		6	LOAD_GLOBAL	0	(_coconut)
		8	LOAD_ATTR	2	(slice)
		10	CALL_METHOD	2	
		12	POP_JUMP_IF_FALSE	48	
353		14	LOAD_FAST	0	(self)
		16	LOAD_ATTR	3	(__class__)
		18	LOAD_FAST	0	(self)
		20	LOAD_ATTR	4	(func)
		22	BUILD_TUPLE	1	
		24	LOAD_CLOSURE	0	(index)
		26	BUILD_TUPLE	1	
		28	LOAD_CONST	1	(<code object <genexpr> at 0x7f6cf32a3f50)
		30	LOAD_CONST	2	('map.__getitem__.<locals>.<genexpr>')
		32	MAKE_FUNCTION	8	(closure)
		34	LOAD_FAST	0	(self)
		36	LOAD_ATTR	5	(iters)
		38	GET_ITER		
		40	CALL_FUNCTION	1	
		42	BUILD_TUPLE_UNPACK_WITH_CALL	2	
		44	CALL_FUNCTION_EX	0	
		46	RETURN_VALUE		
354	>>	48	LOAD_FAST	0	(self)
		50	LOAD_ATTR	4	(func)
		52	LOAD_CLOSURE	0	(index)
		54	BUILD_TUPLE	1	
		56	LOAD_CONST	3	(<code object <genexpr> at 0x7f6cf32a6000)
		58	LOAD_CONST	2	('map.__getitem__.<locals>.<genexpr>')
		60	MAKE_FUNCTION	8	(closure)
		62	LOAD_FAST	0	(self)
		64	LOAD_ATTR	5	(iters)
		66	GET_ITER		
		68	CALL_FUNCTION	1	
		70	CALL_FUNCTION_EX	0	
		72	RETURN_VALUE		

Disassembly of <code object <genexpr> at 0x7f6cf32a3f50, file "<dis>", line 353>:

```

353      0 LOAD_FAST          0 (.0)
      >>  2 FOR_ITER          16 (to 20)
          4 STORE_FAST       1 (i)
          6 LOAD_GLOBAL      0 (_coconut_igetitem)
          8 LOAD_FAST        1 (i)
         10 LOAD_DEREF       0 (index)
         12 CALL_FUNCTION     2
         14 YIELD_VALUE
         16 POP_TOP
         18 JUMP_ABSOLUTE     2
      >>  20 LOAD_CONST      0 (None)
         22 RETURN_VALUE

```

Disassembly of <code object <genexpr> at 0x7f6cf32a6030, file "<dis>", line 354>:

```

354      0 LOAD_FAST          0 (.0)
      >>  2 FOR_ITER          16 (to 20)
          4 STORE_FAST       1 (i)
          6 LOAD_GLOBAL      0 (_coconut_igetitem)
          8 LOAD_FAST        1 (i)
         10 LOAD_DEREF       0 (index)
         12 CALL_FUNCTION     2
         14 YIELD_VALUE
         16 POP_TOP
         18 JUMP_ABSOLUTE     2
      >>  20 LOAD_CONST      0 (None)
         22 RETURN_VALUE

```

Disassembly of <code object __reversed__ at 0x7f6cf32a6240, file "<dis>", line 355>:

```

356      0 LOAD_FAST          0 (self)
          2 LOAD_ATTR         0 (__class__)
          4 LOAD_FAST          0 (self)
          6 LOAD_ATTR         1 (func)
          8 BUILD_TUPLE      1
         10 LOAD_CONST          1 (<code object <genexpr> at 0x7f6cf32a61...)
         12 LOAD_CONST          2 ('map.__reversed__.<locals>.<genexpr>')
         14 MAKE_FUNCTION      0
         16 LOAD_FAST          0 (self)
         18 LOAD_ATTR         2 (iters)

```



```

20 GET_ITER
22 CALL_FUNCTION          1
24 BUILD_TUPLE_UNPACK_WITH_CALL    2
26 CALL_FUNCTION_EX      0
28 RETURN_VALUE

```

Disassembly of <code object <genexpr> at 0x7f6cf32a6190, file "<dis>", line 356>:

```

356      0 LOAD_FAST          0 (.0)
      >>  2 FOR_ITER              14 (to 18)
          4 STORE_FAST         1 (i)
          6 LOAD_GLOBAL        0 (_coconut_reversed)
          8 LOAD_FAST          1 (i)
         10 CALL_FUNCTION        1
         12 YIELD_VALUE
         14 POP_TOP
         16 JUMP_ABSOLUTE        2
      >>  18 LOAD_CONST        0 (None)
         20 RETURN_VALUE

```

Disassembly of <code object __len__ at 0x7f6cf32a63a0, file "<dis>", line 357>:

```

358      0 LOAD_GLOBAL        0 (_coconut)
          2 LOAD_METHOD          1 (min)
          4 LOAD_CONST          1 (<code object <genexpr> at 0x7f6cf32a62f0>)
          6 LOAD_CONST          2 ('map.__len__.__locals__.__len__')
          8 MAKE_FUNCTION        0
         10 LOAD_FAST          0 (self)
         12 LOAD_ATTR          2 (iters)
         14 GET_ITER
         16 CALL_FUNCTION        1
         18 CALL_METHOD        1
         20 RETURN_VALUE

```

Disassembly of <code object <genexpr> at 0x7f6cf32a62f0, file "<dis>", line 358>:

```

358      0 LOAD_FAST          0 (.0)
      >>  2 FOR_ITER              16 (to 20)
          4 STORE_FAST         1 (i)
          6 LOAD_GLOBAL        0 (_coconut)
          8 LOAD_METHOD          1 (len)

```

```

10 LOAD_FAST          1 (i)
12 CALL_METHOD        1
14 YIELD_VALUE
16 POP_TOP
18 JUMP_ABSOLUTE      2
>> 20 LOAD_CONST      0 (None)
22 RETURN_VALUE

```

Disassembly of <code object __repr__ at 0x7f6cf32a6500, file "<dis>", line 359>:

```

360      0 LOAD_CONST      1 ('map(%r, %s)')
      2 LOAD_FAST          0 (self)
      4 LOAD_ATTR          0 (func)
      6 LOAD_CONST        2 (' ', ' ')
      8 LOAD_METHOD        1 (join)
     10 LOAD_CONST        3 (<code object <genexpr> at 0x7f6cf32a6450, file "<dis>", line 360>)
     12 LOAD_CONST        4 ('map.__repr__.<locals>.<genexpr>')
     14 MAKE_FUNCTION      0
     16 LOAD_FAST          0 (self)
     18 LOAD_ATTR          2 (iters)
     20 GET_ITER
     22 CALL_FUNCTION      1
     24 CALL_METHOD        1
     26 BUILD_TUPLE        2
     28 BINARY_MODULO
     30 RETURN_VALUE

```

Disassembly of <code object <genexpr> at 0x7f6cf32a6450, file "<dis>", line 360>:

```

360      0 LOAD_FAST          0 (.0)
>>      2 FOR_ITER          16 (to 20)
      4 STORE_FAST        1 (i)
      6 LOAD_GLOBAL        0 (_coconut)
      8 LOAD_METHOD        1 (repr)
     10 LOAD_FAST          1 (i)
     12 CALL_METHOD        1
     14 YIELD_VALUE
     16 POP_TOP
     18 JUMP_ABSOLUTE      2
>>     20 LOAD_CONST      0 (None)

```

22 RETURN_VALUE

Disassembly of <code object __reduce__ at 0x7f6cf32a65b0, file "<dis>", line 361>:

```
362      0 LOAD_FAST      0 (self)
      2 LOAD_ATTR      0 (__class__)
      4 LOAD_FAST      0 (self)
      6 LOAD_ATTR      1 (func)
      8 BUILD_TUPLE      1
     10 LOAD_FAST      0 (self)
     12 LOAD_ATTR      2 (iters)
     14 BINARY_ADD
     16 BUILD_TUPLE      2
     18 RETURN_VALUE
```

Disassembly of <code object __reduce_ex__ at 0x7f6cf32a6660, file "<dis>", line 362>:

```
364      0 LOAD_FAST      0 (self)
      2 LOAD_METHOD      0 (__reduce__)
      4 CALL_METHOD      0
      6 RETURN_VALUE
```

Disassembly of <code object __copy__ at 0x7f6cf32a6710, file "<dis>", line 365>:

```
366      0 LOAD_FAST      0 (self)
      2 LOAD_ATTR      0 (__class__)
      4 LOAD_FAST      0 (self)
      6 LOAD_ATTR      1 (func)
      8 BUILD_TUPLE      1
     10 LOAD_GLOBAL      2 (_coconut)
     12 LOAD_METHOD      3 (map)
     14 LOAD_GLOBAL      2 (_coconut)
     16 LOAD_ATTR      4 (copy)
     18 LOAD_ATTR      4 (copy)
     20 LOAD_FAST      0 (self)
     22 LOAD_ATTR      5 (iters)
     24 CALL_METHOD      2
     26 BUILD_TUPLE_UNPACK_WITH_CALL      2
     28 CALL_FUNCTION_EX      0
     30 RETURN_VALUE
```

Disassembly of <code object __fmap__ at 0x7f6cf32a67c0, file "<dis>", line 367>:

```

368      0 LOAD_FAST      0 (self)
      2 LOAD_ATTR      0 (__class__)
      4 LOAD_GLOBAL    1 (_coconut_forward_compose)
      6 LOAD_FAST      0 (self)
      8 LOAD_ATTR      2 (func)
     10 LOAD_FAST      1 (func)
     12 CALL_FUNCTION    2
     14 BUILD_TUPLE      1
     16 LOAD_FAST      0 (self)
     18 LOAD_ATTR      3 (iters)
     20 BUILD_TUPLE_UNPACK_WITH_CALL    2
     22 CALL_FUNCTION_EX    0
     24 RETURN_VALUE

```

Disassembly of <code object parallel_map at 0x7f6cf32a6be0, file "<dis>", line 369>:

```

369      0 LOAD_NAME      0 (__name__)
      2 STORE_NAME      1 (__module__)
      4 LOAD_CONST      0 ('parallel_map')
      6 STORE_NAME      2 (__qualname__)

370      8 LOAD_CONST      1 ('Multi-process implementation of map u
     10 STORE_NAME      3 (__doc__)

372     12 LOAD_CONST      2 (())
     14 STORE_NAME      4 (__slots__)

373     16 LOAD_CONST      3 (<code object __iter__ at 0x7f6cf32a687
     18 LOAD_CONST      4 ('parallel_map.__iter__')
     20 MAKE_FUNCTION    0
     22 STORE_NAME      5 (__iter__)

377     24 LOAD_CONST      5 (<code object __repr__ at 0x7f6cf32a6b3
     26 LOAD_CONST      6 ('parallel_map.__repr__')
     28 MAKE_FUNCTION    0
     30 STORE_NAME      6 (__repr__)
     32 LOAD_CONST      7 (None)
     34 RETURN_VALUE

```

Disassembly of <code object __iter__ at 0x7f6cf32a6870, file "<dis>", line 373>:

```

374      0 LOAD_CONST          1 (0)
      2 LOAD_CONST          2 (('ProcessPoolExecutor',))
      4 IMPORT_NAME         0 (concurrent.futures)
      6 IMPORT_FROM        1 (ProcessPoolExecutor)
      8 STORE_FAST         1 (ProcessPoolExecutor)
     10 POP_TOP

375     12 LOAD_FAST          1 (ProcessPoolExecutor)
     14 CALL_FUNCTION       0
     16 SETUP_WITH        46 (to 64)
     18 STORE_FAST         2 (executor)

376     20 LOAD_GLOBAL        2 (_coconut)
     22 LOAD_METHOD        3 (iter)
     24 LOAD_GLOBAL        2 (_coconut)
     26 LOAD_METHOD        4 (list)
     28 LOAD_FAST          2 (executor)
     30 LOAD_ATTR          5 (map)
     32 LOAD_FAST          0 (self)
     34 LOAD_ATTR          6 (func)
     36 BUILD_TUPLE        1
     38 LOAD_FAST          0 (self)
     40 LOAD_ATTR          7 (iters)
     42 BUILD_TUPLE_UNPACK_WITH_CALL    2
     44 CALL_FUNCTION_EX    0
     46 CALL_METHOD        1
     48 CALL_METHOD        1
     50 POP_BLOCK
     52 ROT_TWO
     54 BEGIN_FINALLY
     56 WITH_CLEANUP_START
     58 WITH_CLEANUP_FINISH
     60 POP_FINALLY        0
     62 RETURN_VALUE
>>    64 WITH_CLEANUP_START
     66 WITH_CLEANUP_FINISH

```

```

68 END_FINALLY
70 LOAD_CONST          0 (None)
72 RETURN_VALUE

```

Disassembly of <code object __repr__ at 0x7f6cf32a6b30, file "<dis>", line 377>:

```

378      0 LOAD_CONST          1 ('parallel_')
      2 LOAD_GLOBAL             0 (_coconut_map)
      4 LOAD_METHOD            1 (__repr__)
      6 LOAD_FAST              0 (self)
      8 CALL_METHOD            1
     10 BINARY_ADD
     12 RETURN_VALUE

```

Disassembly of <code object concurrent_map at 0x7f6cf32a6df0, file "<dis>", line 378>:

```

379      0 LOAD_NAME             0 (__name__)
      2 STORE_NAME              1 (__module__)
      4 LOAD_CONST             0 ('concurrent_map')
      6 STORE_NAME              2 (__qualname__)

380      8 LOAD_CONST            1 ('Multi-thread implementation of map us
     10 STORE_NAME              3 (__doc__)

381     12 LOAD_CONST            2 (())
     14 STORE_NAME              4 (__slots__)

382     16 LOAD_CONST            3 (<code object __iter__ at 0x7f6cf32a6c90>)
     18 LOAD_CONST            4 ('concurrent_map.__iter__')
     20 MAKE_FUNCTION          0
     22 STORE_NAME              5 (__iter__)

387     24 LOAD_CONST            5 (<code object __repr__ at 0x7f6cf32a6d40>)
     26 LOAD_CONST            6 ('concurrent_map.__repr__')
     28 MAKE_FUNCTION          0
     30 STORE_NAME              6 (__repr__)
     32 LOAD_CONST            7 (None)
     34 RETURN_VALUE

```

Disassembly of <code object __iter__ at 0x7f6cf32a6c90, file "<dis>", line 382>:

383	0 LOAD_CONST	1 (0)
	2 LOAD_CONST	2 (('ThreadPoolExecutor',))
	4 IMPORT_NAME	0 (concurrent.futures)
	6 IMPORT_FROM	1 (ThreadPoolExecutor)
	8 STORE_FAST	1 (ThreadPoolExecutor)
	10 POP_TOP	
384	12 LOAD_CONST	1 (0)
	14 LOAD_CONST	3 (('cpu_count',))
	16 IMPORT_NAME	2 (multiprocessing)
	18 IMPORT_FROM	3 (cpu_count)
	20 STORE_FAST	2 (cpu_count)
	22 POP_TOP	
385	24 LOAD_FAST	1 (ThreadPoolExecutor)
	26 LOAD_FAST	2 (cpu_count)
	28 CALL_FUNCTION	0
	30 LOAD_CONST	4 (5)
	32 BINARY_MULTIPLY	
	34 CALL_FUNCTION	1
	36 SETUP_WITH	46 (to 84)
	38 STORE_FAST	3 (executor)
386	40 LOAD_GLOBAL	4 (_coconut)
	42 LOAD_METHOD	5 (iter)
	44 LOAD_GLOBAL	4 (_coconut)
	46 LOAD_METHOD	6 (list)
	48 LOAD_FAST	3 (executor)
	50 LOAD_ATTR	7 (map)
	52 LOAD_FAST	0 (self)
	54 LOAD_ATTR	8 (func)
	56 BUILD_TUPLE	1
	58 LOAD_FAST	0 (self)
	60 LOAD_ATTR	9 (iters)
	62 BUILD_TUPLE_UNPACK_WITH_CALL	2
	64 CALL_FUNCTION_EX	0
	66 CALL_METHOD	1
	68 CALL_METHOD	1

```

70 POP_BLOCK
72 ROT_TWO
74 BEGIN_FINALLY
76 WITH_CLEANUP_START
78 WITH_CLEANUP_FINISH
80 POP_FINALLY          0
82 RETURN_VALUE
>> 84 WITH_CLEANUP_START
86 WITH_CLEANUP_FINISH
88 END_FINALLY
90 LOAD_CONST            0 (None)
92 RETURN_VALUE

```

Disassembly of <code object __repr__ at 0x7f6cf32a6d40, file "<dis>", line 387>:

```

388      0 LOAD_CONST          1 ('concurrent_')
      2 LOAD_GLOBAL             0 (_coconut_map)
      4 LOAD_METHOD            1 (__repr__)
      6 LOAD_FAST              0 (self)
      8 CALL_METHOD            1
     10 BINARY_ADD
     12 RETURN_VALUE

```

Disassembly of <code object filter at 0x7f6cf32ab3a0, file "<dis>", line 389>:

```

389      0 LOAD_NAME             0 (__name__)
      2 STORE_NAME              1 (__module__)
      4 LOAD_CONST             0 ('filter')
      6 STORE_NAME             2 (__qualname__)

390      8 LOAD_CONST             1 (('func', 'iter'))
     10 STORE_NAME             3 (__slots__)

391     12 LOAD_NAME             4 (hasattr)
     14 LOAD_NAME             5 (_coconut)
     16 LOAD_ATTR              6 (filter)
     18 LOAD_CONST             2 ('__doc__')
     20 CALL_FUNCTION          2
     22 POP_JUMP_IF_FALSE     32

```


392		24 LOAD_NAME	5 (_coconut)
		26 LOAD_ATTR	6 (filter)
		28 LOAD_ATTR	7 (__doc__)
		30 STORE_NAME	7 (__doc__)
393	>>	32 LOAD_CONST	3 (<code object __new__ at 0x7f6cf32a6ea0)
		34 LOAD_CONST	4 ('filter.__new__')
		36 MAKE_FUNCTION	0
		38 STORE_NAME	8 (__new__)
398		40 LOAD_CONST	5 (<code object __reversed__ at 0x7f6cf32a6ea0)
		42 LOAD_CONST	6 ('filter.__reversed__')
		44 MAKE_FUNCTION	0
		46 STORE_NAME	9 (__reversed__)
400		48 LOAD_CONST	7 (<code object __repr__ at 0x7f6cf32a6ea0)
		50 LOAD_CONST	8 ('filter.__repr__')
		52 MAKE_FUNCTION	0
		54 STORE_NAME	10 (__repr__)
402		56 LOAD_CONST	9 (<code object __reduce__ at 0x7f6cf32a6ea0)
		58 LOAD_CONST	10 ('filter.__reduce__')
		60 MAKE_FUNCTION	0
		62 STORE_NAME	11 (__reduce__)
404		64 LOAD_CONST	11 (<code object __reduce_ex__ at 0x7f6cf32a6ea0)
		66 LOAD_CONST	12 ('filter.__reduce_ex__')
		68 MAKE_FUNCTION	0
		70 STORE_NAME	12 (__reduce_ex__)
406		72 LOAD_CONST	13 (<code object __copy__ at 0x7f6cf32a6ea0)
		74 LOAD_CONST	14 ('filter.__copy__')
		76 MAKE_FUNCTION	0
		78 STORE_NAME	13 (__copy__)
408		80 LOAD_CONST	15 (<code object __fmap__ at 0x7f6cf32a6ea0)
		82 LOAD_CONST	16 ('filter.__fmap__')
		84 MAKE_FUNCTION	0

86	STORE_NAME	14	(__fmap__)
88	LOAD_CONST	17	(None)
90	RETURN_VALUE		

Disassembly of <code object __new__ at 0x7f6cf32a6ea0, file "<dis>", line 393>:

394	0	LOAD_GLOBAL	0	(_coconut)
	2	LOAD_ATTR	1	(filter)
	4	LOAD_METHOD	2	(__new__)
	6	LOAD_FAST	0	(cls)
	8	LOAD_FAST	1	(function)
	10	LOAD_FAST	2	(iterable)
	12	CALL_METHOD	3	
	14	STORE_FAST	3	(new_filter)
395	16	LOAD_FAST	1	(function)
	18	LOAD_FAST	3	(new_filter)
	20	STORE_ATTR	3	(func)
396	22	LOAD_FAST	2	(iterable)
	24	LOAD_FAST	3	(new_filter)
	26	STORE_ATTR	4	(iter)
397	28	LOAD_FAST	3	(new_filter)
	30	RETURN_VALUE		

Disassembly of <code object __reversed__ at 0x7f6cf32a6f50, file "<dis>", line 398>:

399	0	LOAD_FAST	0	(self)
	2	LOAD_METHOD	0	(__class__)
	4	LOAD_FAST	0	(self)
	6	LOAD_ATTR	1	(func)
	8	LOAD_GLOBAL	2	(_coconut_reversed)
	10	LOAD_FAST	0	(self)
	12	LOAD_ATTR	3	(iter)
	14	CALL_FUNCTION	1	
	16	CALL_METHOD	2	
	18	RETURN_VALUE		

Disassembly of <code object __repr__ at 0x7f6cf32ab030, file "<dis>", line 400>:

```

401          0 LOAD_CONST          1 ('filter(%r, %r)')
          2 LOAD_FAST              0 (self)
          4 LOAD_ATTR              0 (func)
          6 LOAD_FAST              0 (self)
          8 LOAD_ATTR              1 (iter)
         10 BUILD_TUPLE            2
         12 BINARY_MODULO
         14 RETURN_VALUE

```

Disassembly of <code object __reduce__ at 0x7f6cf32ab0e0, file "<dis>", line 402>:

```

403          0 LOAD_FAST              0 (self)
          2 LOAD_ATTR              0 (__class__)
          4 LOAD_FAST              0 (self)
          6 LOAD_ATTR              1 (func)
          8 LOAD_FAST              0 (self)
         10 LOAD_ATTR              2 (iter)
         12 BUILD_TUPLE            2
         14 BUILD_TUPLE            2
         16 RETURN_VALUE

```

Disassembly of <code object __reduce_ex__ at 0x7f6cf32ab190, file "<dis>", line 404>:

```

405          0 LOAD_FAST              0 (self)
          2 LOAD_METHOD            0 (__reduce__)
          4 CALL_METHOD            0
          6 RETURN_VALUE

```

Disassembly of <code object __copy__ at 0x7f6cf32ab240, file "<dis>", line 406>:

```

407          0 LOAD_FAST              0 (self)
          2 LOAD_METHOD            0 (__class__)
          4 LOAD_FAST              0 (self)
          6 LOAD_ATTR              1 (func)
          8 LOAD_GLOBAL            2 (_coconut)
         10 LOAD_ATTR              3 (copy)
         12 LOAD_METHOD            3 (copy)
         14 LOAD_FAST              0 (self)
         16 LOAD_ATTR              4 (iter)
         18 CALL_METHOD            1
         20 CALL_METHOD            2

```

22 RETURN_VALUE

Disassembly of <code object __fmap__ at 0x7f6cf32ab2f0, file "<dis>", line 408>:

```
409      0 LOAD_GLOBAL      0 (_coconut_map)
      2 LOAD_FAST          1 (func)
      4 LOAD_FAST          0 (self)
      6 CALL_FUNCTION      2
      8 RETURN_VALUE
```

Disassembly of <code object zip at 0x7f6cf32abea0, file "<dis>", line 410>:

```
410      0 LOAD_NAME          0 (__name__)
      2 STORE_NAME          1 (__module__)
      4 LOAD_CONST          0 ('zip')
      6 STORE_NAME          2 (__qualname__)
```

```
411      8 LOAD_CONST          1 (('iters',))
     10 STORE_NAME          3 (__slots__)
```

```
412     12 LOAD_NAME          4 (hasattr)
     14 LOAD_NAME          5 (_coconut)
     16 LOAD_ATTR          6 (zip)
     18 LOAD_CONST          2 ('__doc__')
     20 CALL_FUNCTION      2
     22 POP_JUMP_IF_FALSE  32
```

```
413     24 LOAD_NAME          5 (_coconut)
     26 LOAD_ATTR          6 (zip)
     28 LOAD_ATTR          7 (__doc__)
     30 STORE_NAME          7 (__doc__)
```

```
414    >>  32 LOAD_CONST          3 (<code object __new__ at 0x7f6cf32ab450>)
     34 LOAD_CONST          4 ('zip.__new__')
     36 MAKE_FUNCTION      0
     38 STORE_NAME          8 (__new__)
```

```
418     40 LOAD_CONST          5 (<code object __getitem__ at 0x7f6cf32ab450>)
     42 LOAD_CONST          6 ('zip.__getitem__')
     44 MAKE_FUNCTION      0
```

	46 STORE_NAME	9 (__getitem__)
422	48 LOAD_CONST	7 (<code object __reversed__ at 0x7f6cf32ab920)
	50 LOAD_CONST	8 ('zip.__reversed__')
	52 MAKE_FUNCTION	0
	54 STORE_NAME	10 (__reversed__)
424	56 LOAD_CONST	9 (<code object __len__ at 0x7f6cf32ab920)
	58 LOAD_CONST	10 ('zip.__len__')
	60 MAKE_FUNCTION	0
	62 STORE_NAME	11 (__len__)
426	64 LOAD_CONST	11 (<code object __repr__ at 0x7f6cf32aba80)
	66 LOAD_CONST	12 ('zip.__repr__')
	68 MAKE_FUNCTION	0
	70 STORE_NAME	12 (__repr__)
428	72 LOAD_CONST	13 (<code object __reduce__ at 0x7f6cf32ab920)
	74 LOAD_CONST	14 ('zip.__reduce__')
	76 MAKE_FUNCTION	0
	78 STORE_NAME	13 (__reduce__)
430	80 LOAD_CONST	15 (<code object __reduce_ex__ at 0x7f6cf32ab920)
	82 LOAD_CONST	16 ('zip.__reduce_ex__')
	84 MAKE_FUNCTION	0
	86 STORE_NAME	14 (__reduce_ex__)
432	88 LOAD_CONST	17 (<code object __copy__ at 0x7f6cf32abc90)
	90 LOAD_CONST	18 ('zip.__copy__')
	92 MAKE_FUNCTION	0
	94 STORE_NAME	15 (__copy__)
434	96 LOAD_CONST	19 (<code object __fmap__ at 0x7f6cf32abd40)
	98 LOAD_CONST	20 ('zip.__fmap__')
	100 MAKE_FUNCTION	0
	102 STORE_NAME	16 (__fmap__)
	104 LOAD_CONST	21 (None)
	106 RETURN_VALUE	

Disassembly of <code object __new__ at 0x7f6cf32ab450, file "<dis>", line 414>:

```

415      0 LOAD_GLOBAL          0 (_coconut)
      2 LOAD_ATTR                1 (zip)
      4 LOAD_ATTR                2 (__new__)
      6 LOAD_FAST                0 (cls)
      8 BUILD_TUPLE              1
     10 LOAD_FAST                1 (iterables)
     12 BUILD_TUPLE_UNPACK_WITH_CALL    2
     14 CALL_FUNCTION_EX          0
     16 STORE_FAST              2 (new_zip)

416     18 LOAD_FAST              1 (iterables)
     20 LOAD_FAST              2 (new_zip)
     22 STORE_ATTR              3 (iters)

417     24 LOAD_FAST              2 (new_zip)
     26 RETURN_VALUE

```

Disassembly of <code object __getitem__ at 0x7f6cf32ab660, file "<dis>", line 418>

```

419      0 LOAD_GLOBAL          0 (_coconut)
      2 LOAD_METHOD              1 (isinstance)
      4 LOAD_DEREF              0 (index)
      6 LOAD_GLOBAL          0 (_coconut)
      8 LOAD_ATTR                2 (slice)
     10 CALL_METHOD              2
     12 POP_JUMP_IF_FALSE       40

420     14 LOAD_FAST              0 (self)
     16 LOAD_ATTR                3 (__class__)
     18 LOAD_CLOSURE             0 (index)
     20 BUILD_TUPLE              1
     22 LOAD_CONST               1 (<code object <genexpr> at 0x7f6cf32ab5
     24 LOAD_CONST               2 ('zip.__getitem__.<locals>.<genexpr>')
     26 MAKE_FUNCTION             8 (closure)
     28 LOAD_FAST              0 (self)
     30 LOAD_ATTR                4 (iters)
     32 GET_ITER

```

```

34 CALL_FUNCTION          1
36 CALL_FUNCTION_EX       0
38 RETURN_VALUE

421  >> 40 LOAD_GLOBAL      0 (_coconut)
      42 LOAD_METHOD        5 (tuple)
      44 LOAD_CLOSURE        0 (index)
      46 BUILD_TUPLE        1
      48 LOAD_CONST          3 (<code object <genexpr> at 0x7f6cf32ab5
      50 LOAD_CONST          2 ('zip.__getitem__.<locals>.<genexpr>')
      52 MAKE_FUNCTION        8 (closure)
      54 LOAD_FAST           0 (self)
      56 LOAD_ATTR           4 (iters)
      58 GET_ITER
      60 CALL_FUNCTION        1
      62 CALL_METHOD         1
      64 RETURN_VALUE

```

Disassembly of <code object <genexpr> at 0x7f6cf32ab500, file "<dis>", line 420>:

```

420      0 LOAD_FAST          0 (.0)
      >>  2 FOR_ITER           16 (to 20)
      4 STORE_FAST          1 (i)
      6 LOAD_GLOBAL          0 (_coconut_igetitem)
      8 LOAD_FAST            1 (i)
     10 LOAD_DEREF            0 (index)
     12 CALL_FUNCTION         2
     14 YIELD_VALUE
     16 POP_TOP
     18 JUMP_ABSOLUTE         2
      >> 20 LOAD_CONST          0 (None)
     22 RETURN_VALUE

```

Disassembly of <code object <genexpr> at 0x7f6cf32ab5b0, file "<dis>", line 421>:

```

421      0 LOAD_FAST          0 (.0)
      >>  2 FOR_ITER           16 (to 20)
      4 STORE_FAST          1 (i)
      6 LOAD_GLOBAL          0 (_coconut_igetitem)
      8 LOAD_FAST            1 (i)

```

```

10 LOAD_DEREF          0 (index)
12 CALL_FUNCTION       2
14 YIELD_VALUE
16 POP_TOP
18 JUMP_ABSOLUTE       2
>> 20 LOAD_CONST        0 (None)
22 RETURN_VALUE

```

Disassembly of <code object __reversed__ at 0x7f6cf32ab7c0, file "<dis>", line 422>:

```

423      0 LOAD_FAST      0 (self)
      2 LOAD_ATTR         0 (__class__)
      4 LOAD_CONST       1 (<code object <genexpr> at 0x7f6cf32ab7c0, file "<dis>", line 422>)
      6 LOAD_CONST       2 ('zip.__reversed__.<locals>.<genexpr>')
      8 MAKE_FUNCTION    0
     10 LOAD_FAST      0 (self)
     12 LOAD_ATTR         1 (iters)
     14 GET_ITER
     16 CALL_FUNCTION     1
     18 CALL_FUNCTION_EX  0
     20 RETURN_VALUE

```

Disassembly of <code object <genexpr> at 0x7f6cf32ab710, file "<dis>", line 423>:

```

423      0 LOAD_FAST      0 (.0)
>>     2 FOR_ITER        14 (to 18)
      4 STORE_FAST       1 (i)
      6 LOAD_GLOBAL     0 (_coconut_reversed)
      8 LOAD_FAST       1 (i)
     10 CALL_FUNCTION     1
     12 YIELD_VALUE
     14 POP_TOP
     16 JUMP_ABSOLUTE    2
>>     18 LOAD_CONST        0 (None)
     20 RETURN_VALUE

```

Disassembly of <code object __len__ at 0x7f6cf32ab920, file "<dis>", line 424>:

```

425      0 LOAD_GLOBAL     0 (_coconut)
      2 LOAD_METHOD      1 (min)
      4 LOAD_CONST       1 (<code object <genexpr> at 0x7f6cf32ab710, file "<dis>", line 423>)

```


6	LOAD_CONST	2 ('zip.__len__.<locals>.<genexpr>')
8	MAKE_FUNCTION	0
10	LOAD_FAST	0 (self)
12	LOAD_ATTR	2 (iters)
14	GET_ITER	
16	CALL_FUNCTION	1
18	CALL_METHOD	1
20	RETURN_VALUE	

Disassembly of <code object <genexpr> at 0x7f6cf32ab870, file "<dis>", line 425>:

425	0	LOAD_FAST	0 (.0)
>>	2	FOR_ITER	16 (to 20)
	4	STORE_FAST	1 (i)
	6	LOAD_GLOBAL	0 (_coconut)
	8	LOAD_METHOD	1 (len)
	10	LOAD_FAST	1 (i)
	12	CALL_METHOD	1
	14	YIELD_VALUE	
	16	POP_TOP	
	18	JUMP_ABSOLUTE	2
>>	20	LOAD_CONST	0 (None)
	22	RETURN_VALUE	

Disassembly of <code object __repr__ at 0x7f6cf32aba80, file "<dis>", line 426>:

427	0	LOAD_CONST	1 ('zip(%s)')
	2	LOAD_CONST	2 ('', '')
	4	LOAD_METHOD	0 (join)
	6	LOAD_CONST	3 (<code object <genexpr> at 0x7f6cf32ab9...
	8	LOAD_CONST	4 ('zip.__repr__.<locals>.<genexpr>')
	10	MAKE_FUNCTION	0
	12	LOAD_FAST	0 (self)
	14	LOAD_ATTR	1 (iters)
	16	GET_ITER	
	18	CALL_FUNCTION	1
	20	CALL_METHOD	1
	22	BUILD_TUPLE	1
	24	BINARY_MODULO	
	26	RETURN_VALUE	

Disassembly of <code object <genexpr> at 0x7f6cf32ab9d0, file "<dis>", line 427>:

```
427      0 LOAD_FAST          0 (.0)
      >>  2 FOR_ITER            16 (to 20)
          4 STORE_FAST        1 (i)
          6 LOAD_GLOBAL      0 (_coconut)
          8 LOAD_METHOD        1 (repr)
         10 LOAD_FAST          1 (i)
         12 CALL_METHOD        1
         14 YIELD_VALUE
         16 POP_TOP
         18 JUMP_ABSOLUTE      2
      >>  20 LOAD_CONST        0 (None)
         22 RETURN_VALUE
```

Disassembly of <code object __reduce__ at 0x7f6cf32abb30, file "<dis>", line 428>:

```
429      0 LOAD_FAST          0 (self)
          2 LOAD_ATTR          0 (__class__)
          4 LOAD_FAST          0 (self)
          6 LOAD_ATTR          1 (iters)
          8 BUILD_TUPLE        2
         10 RETURN_VALUE
```

Disassembly of <code object __reduce_ex__ at 0x7f6cf32abbe0, file "<dis>", line 430>:

```
431      0 LOAD_FAST          0 (self)
          2 LOAD_METHOD          0 (__reduce__)
          4 CALL_METHOD        0
          6 RETURN_VALUE
```

Disassembly of <code object __copy__ at 0x7f6cf32abc90, file "<dis>", line 432>:

```
433      0 LOAD_FAST          0 (self)
          2 LOAD_ATTR          0 (__class__)
          4 LOAD_GLOBAL          1 (_coconut)
          6 LOAD_METHOD        2 (map)
          8 LOAD_GLOBAL          1 (_coconut)
         10 LOAD_ATTR            3 (copy)
         12 LOAD_ATTR            3 (copy)
         14 LOAD_FAST          0 (self)
```

16	LOAD_ATTR	4 (iters)
18	CALL_METHOD	2
20	CALL_FUNCTION_EX	0
22	RETURN_VALUE	

Disassembly of <code object __fmap__ at 0x7f6cf32abd40, file "<dis>", line 434>:

435	0	LOAD_GLOBAL	0 (_coconut_map)
	2	LOAD_FAST	1 (func)
	4	LOAD_FAST	0 (self)
	6	CALL_FUNCTION	2
	8	RETURN_VALUE	

Disassembly of <code object enumerate at 0x7f6cf32ac660, file "<dis>", line 436>:

436	0	LOAD_NAME	0 (__name__)
	2	STORE_NAME	1 (__module__)
	4	LOAD_CONST	0 ('enumerate')
	6	STORE_NAME	2 (__qualname__)
437	8	LOAD_CONST	1 (('iter', 'start'))
	10	STORE_NAME	3 (__slots__)

438	12	LOAD_NAME	4 (hasattr)
	14	LOAD_NAME	5 (_coconut)
	16	LOAD_ATTR	6 (enumerate)
	18	LOAD_CONST	2 ('__doc__')
	20	CALL_FUNCTION	2
	22	POP_JUMP_IF_FALSE	32

439	24	LOAD_NAME	5 (_coconut)
	26	LOAD_ATTR	6 (enumerate)
	28	LOAD_ATTR	7 (__doc__)
	30	STORE_NAME	7 (__doc__)

440	>>	32	LOAD_CONST	21 ((0,))
		34	LOAD_CONST	4 (<code object __new__ at 0x7f6cf32abdf0)
		36	LOAD_CONST	5 ('enumerate.__new__')
		38	MAKE_FUNCTION	1 (defaults)
		40	STORE_NAME	8 (__new__)

445	42 LOAD_CONST	6 (<code object __getitem__ at 0x7f6cf32a
	44 LOAD_CONST	7 ('enumerate.__getitem__')
	46 MAKE_FUNCTION	0
	48 STORE_NAME	9 (__getitem__)
449	50 LOAD_CONST	8 (<code object __len__ at 0x7f6cf32ac0e0
	52 LOAD_CONST	9 ('enumerate.__len__')
	54 MAKE_FUNCTION	0
	56 STORE_NAME	10 (__len__)
451	58 LOAD_CONST	10 (<code object __repr__ at 0x7f6cf32ac24
	60 LOAD_CONST	11 ('enumerate.__repr__')
	62 MAKE_FUNCTION	0
	64 STORE_NAME	11 (__repr__)
453	66 LOAD_CONST	12 (<code object __reduce__ at 0x7f6cf32ac
	68 LOAD_CONST	13 ('enumerate.__reduce__')
	70 MAKE_FUNCTION	0
	72 STORE_NAME	12 (__reduce__)
455	74 LOAD_CONST	14 (<code object __reduce_ex__ at 0x7f6cf3
	76 LOAD_CONST	15 ('enumerate.__reduce_ex__')
	78 MAKE_FUNCTION	0
	80 STORE_NAME	13 (__reduce_ex__)
457	82 LOAD_CONST	16 (<code object __copy__ at 0x7f6cf32ac45
	84 LOAD_CONST	17 ('enumerate.__copy__')
	86 MAKE_FUNCTION	0
	88 STORE_NAME	14 (__copy__)
459	90 LOAD_CONST	18 (<code object __fmap__ at 0x7f6cf32ac50
	92 LOAD_CONST	19 ('enumerate.__fmap__')
	94 MAKE_FUNCTION	0
	96 STORE_NAME	15 (__fmap__)
	98 LOAD_CONST	20 (None)
	100 RETURN_VALUE	

Disassembly of <code object __new__ at 0x7f6cf32abdf0, file "<dis>", line 440>:

441	0	LOAD_GLOBAL	0	(_coconut)
	2	LOAD_ATTR	1	(enumerate)
	4	LOAD_METHOD	2	(__new__)
	6	LOAD_FAST	0	(cls)
	8	LOAD_FAST	1	(iterable)
	10	LOAD_FAST	2	(start)
	12	CALL_METHOD	3	
	14	STORE_FAST	3	(new_enumerate)
442	16	LOAD_FAST	1	(iterable)
	18	LOAD_FAST	3	(new_enumerate)
	20	STORE_ATTR	3	(iter)
443	22	LOAD_FAST	2	(start)
	24	LOAD_FAST	3	(new_enumerate)
	26	STORE_ATTR	4	(start)
444	28	LOAD_FAST	3	(new_enumerate)
	30	RETURN_VALUE		

Disassembly of <code object __getitem__ at 0x7f6cf32ac190, file "<dis>", line 445>:

446	0	LOAD_GLOBAL	0	(_coconut)
	2	LOAD_METHOD	1	(isinstance)
	4	LOAD_FAST	1	(index)
	6	LOAD_GLOBAL	0	(_coconut)
	8	LOAD_ATTR	2	(slice)
	10	CALL_METHOD	2	
	12	POP_JUMP_IF_FALSE	82	
447	14	LOAD_FAST	0	(self)
	16	LOAD_METHOD	3	(__class__)
	18	LOAD_GLOBAL	4	(_coconut_igetitem)
	20	LOAD_FAST	0	(self)
	22	LOAD_ATTR	5	(iter)
	24	LOAD_FAST	1	(index)
	26	CALL_FUNCTION	2	
	28	LOAD_FAST	0	(self)

	30	LOAD_ATTR	6 (start)	
	32	LOAD_FAST	1 (index)	
	34	LOAD_ATTR	6 (start)	
	36	LOAD_CONST	0 (None)	
	38	COMPARE_OP	8 (is)	
	40	POP_JUMP_IF_FALSE	46	
	42	LOAD_CONST	1 (0)	
	44	JUMP_FORWARD	30 (to 76)	
>>	46	LOAD_FAST	1 (index)	
	48	LOAD_ATTR	6 (start)	
	50	LOAD_CONST	1 (0)	
	52	COMPARE_OP	5 (>=)	
	54	POP_JUMP_IF_FALSE	62	
	56	LOAD_FAST	1 (index)	
	58	LOAD_ATTR	6 (start)	
	60	JUMP_FORWARD	14 (to 76)	
>>	62	LOAD_GLOBAL	7 (len)	
	64	LOAD_FAST	0 (self)	
	66	LOAD_ATTR	5 (iter)	
	68	CALL_FUNCTION	1	
	70	LOAD_FAST	1 (index)	
	72	LOAD_ATTR	6 (start)	
	74	BINARY_ADD		
>>	76	BINARY_ADD		
	78	CALL_METHOD	2	
	80	RETURN_VALUE		
448	>>	82	LOAD_FAST	0 (self)
		84	LOAD_ATTR	6 (start)
		86	LOAD_FAST	1 (index)
		88	BINARY_ADD	
		90	LOAD_GLOBAL	4 (_coconut_igetitem)
		92	LOAD_FAST	0 (self)
		94	LOAD_ATTR	5 (iter)
		96	LOAD_FAST	1 (index)
		98	CALL_FUNCTION	2
		100	BUILD_TUPLE	2
		102	RETURN_VALUE	

```

Disassembly of <code object __len__ at 0x7f6cf32ac0e0, file "<dis>", line 449>:
450          0 LOAD_GLOBAL          0 (_coconut)
          2 LOAD_METHOD          1 (len)
          4 LOAD_FAST           0 (self)
          6 LOAD_ATTR         2 (iter)
          8 CALL_METHOD        1
         10 RETURN_VALUE

```

```

Disassembly of <code object __repr__ at 0x7f6cf32ac240, file "<dis>", line 451>:
452          0 LOAD_CONST          1 ('enumerate(%r, %r)')
          2 LOAD_FAST           0 (self)
          4 LOAD_ATTR         0 (iter)
          6 LOAD_FAST           0 (self)
          8 LOAD_ATTR         1 (start)
         10 BUILD_TUPLE          2
         12 BINARY_MODULO
         14 RETURN_VALUE

```

```

Disassembly of <code object __reduce__ at 0x7f6cf32ac2f0, file "<dis>", line 453>:
454          0 LOAD_FAST           0 (self)
          2 LOAD_ATTR         0 (__class__)
          4 LOAD_FAST           0 (self)
          6 LOAD_ATTR         1 (iter)
          8 LOAD_FAST           0 (self)
         10 LOAD_ATTR         2 (start)
         12 BUILD_TUPLE          2
         14 BUILD_TUPLE          2
         16 RETURN_VALUE

```

```

Disassembly of <code object __reduce_ex__ at 0x7f6cf32ac3a0, file "<dis>", line 455>:
456          0 LOAD_FAST           0 (self)
          2 LOAD_METHOD          0 (__reduce__)
          4 CALL_METHOD          0
          6 RETURN_VALUE

```

```

Disassembly of <code object __copy__ at 0x7f6cf32ac450, file "<dis>", line 457>:
458          0 LOAD_FAST           0 (self)

```

2	LOAD_METHOD	0	(__class__)
4	LOAD_GLOBAL	1	(_coconut)
6	LOAD_ATTR	2	(copy)
8	LOAD_METHOD	2	(copy)
10	LOAD_FAST	0	(self)
12	LOAD_ATTR	3	(iter)
14	CALL_METHOD	1	
16	LOAD_FAST	0	(self)
18	LOAD_ATTR	4	(start)
20	CALL_METHOD	2	
22	RETURN_VALUE		

Disassembly of <code object __fmap__ at 0x7f6cf32ac500, file "<dis>", line 459>:

460	0	LOAD_GLOBAL	0	(_coconut_map)
	2	LOAD_FAST	1	(func)
	4	LOAD_FAST	0	(self)
	6	CALL_FUNCTION	2	
	8	RETURN_VALUE		

Disassembly of <code object count at 0x7f6cf32b1190, file "<dis>", line 461>:

461	0	LOAD_NAME	0	(__name__)
	2	STORE_NAME	1	(__module__)
	4	LOAD_CONST	0	('count')
	6	STORE_NAME	2	(__qualname__)
462	8	LOAD_CONST	1	('count(start, step) returns an infinit
	10	STORE_NAME	3	(__doc__)
464	12	LOAD_CONST	2	((('start', 'step'))
	14	STORE_NAME	4	(__slots__)
465	16	LOAD_CONST	32	((0, 1))
	18	LOAD_CONST	5	(<code object __init__ at 0x7f6cf32ac5b
	20	LOAD_CONST	6	('count.__init__')
	22	MAKE_FUNCTION	1	(defaults)
	24	STORE_NAME	5	(__init__)
468	26	LOAD_CONST	7	(<code object __iter__ at 0x7f6cf32ac87

	28 LOAD_CONST	8 ('count.__iter__')
	30 MAKE_FUNCTION	0
	32 STORE_NAME	6 (__iter__)
473	34 LOAD_CONST	9 (<code object __contains__ at 0x7f6cf32
	36 LOAD_CONST	10 ('count.__contains__')
	38 MAKE_FUNCTION	0
	40 STORE_NAME	7 (__contains__)
479	42 LOAD_CONST	11 (<code object __getitem__ at 0x7f6cf32a
	44 LOAD_CONST	12 ('count.__getitem__')
	46 MAKE_FUNCTION	0
	48 STORE_NAME	8 (__getitem__)
494	50 LOAD_CONST	13 (<code object count at 0x7f6cf32aca80,
	52 LOAD_CONST	14 ('count.count')
	54 MAKE_FUNCTION	0
	56 STORE_NAME	9 (count)
499	58 LOAD_CONST	15 (<code object index at 0x7f6cf32acb30,
	60 LOAD_CONST	16 ('count.index')
	62 MAKE_FUNCTION	0
	64 STORE_NAME	10 (index)
504	66 LOAD_CONST	17 (<code object __reversed__ at 0x7f6cf32
	68 LOAD_CONST	18 ('count.__reversed__')
	70 MAKE_FUNCTION	0
	72 STORE_NAME	11 (__reversed__)
508	74 LOAD_CONST	19 (<code object __repr__ at 0x7f6cf32acc9
	76 LOAD_CONST	20 ('count.__repr__')
	78 MAKE_FUNCTION	0
	80 STORE_NAME	12 (__repr__)
510	82 LOAD_CONST	21 (<code object __hash__ at 0x7f6cf32acd4
	84 LOAD_CONST	22 ('count.__hash__')
	86 MAKE_FUNCTION	0
	88 STORE_NAME	13 (__hash__)

512	90 LOAD_CONST	23 (<code object __reduce__ at 0x7f6cf32ac
	92 LOAD_CONST	24 ('count.__reduce__')
	94 MAKE_FUNCTION	0
	96 STORE_NAME	14 (__reduce__)
514	98 LOAD_CONST	25 (<code object __copy__ at 0x7f6cf32acea
	100 LOAD_CONST	26 ('count.__copy__')
	102 MAKE_FUNCTION	0
	104 STORE_NAME	15 (__copy__)
516	106 LOAD_CONST	27 (<code object __eq__ at 0x7f6cf32acf50,
	108 LOAD_CONST	28 ('count.__eq__')
	110 MAKE_FUNCTION	0
	112 STORE_NAME	16 (__eq__)
518	114 LOAD_CONST	29 (<code object __fmap__ at 0x7f6cf32b103
	116 LOAD_CONST	30 ('count.__fmap__')
	118 MAKE_FUNCTION	0
	120 STORE_NAME	17 (__fmap__)
	122 LOAD_CONST	31 (None)
	124 RETURN_VALUE	

Disassembly of <code object __init__ at 0x7f6cf32ac5b0, file "<dis>", line 465>:

466	0 LOAD_FAST	1 (start)
	2 LOAD_FAST	0 (self)
	4 STORE_ATTR	0 (start)
467	6 LOAD_FAST	2 (step)
	8 LOAD_FAST	0 (self)
	10 STORE_ATTR	1 (step)
	12 LOAD_CONST	0 (None)
	14 RETURN_VALUE	

Disassembly of <code object __iter__ at 0x7f6cf32ac870, file "<dis>", line 468>:

470	>>	0 LOAD_FAST	0 (self)
		2 LOAD_ATTR	0 (start)
		4 YIELD_VALUE	

	6	POP_TOP	
471	8	LOAD_FAST	0 (self)
	10	LOAD_ATTR	1 (step)
	12	POP_JUMP_IF_FALSE	0
472	14	LOAD_FAST	0 (self)
	16	DUP_TOP	
	18	LOAD_ATTR	0 (start)
	20	LOAD_FAST	0 (self)
	22	LOAD_ATTR	1 (step)
	24	INPLACE_ADD	
	26	ROT_TWO	
	28	STORE_ATTR	0 (start)
	30	JUMP_ABSOLUTE	0
	32	LOAD_CONST	0 (None)
	34	RETURN_VALUE	

Disassembly of <code object __contains__ at 0x7f6cf32ac920, file "<dis>", line 473

474	0	LOAD_FAST	0 (self)
	2	LOAD_ATTR	0 (step)
	4	POP_JUMP_IF_TRUE	16
475	6	LOAD_FAST	1 (elem)
	8	LOAD_FAST	0 (self)
	10	LOAD_ATTR	1 (start)
	12	COMPARE_OP	2 (==)
	14	RETURN_VALUE	
476	>> 16	LOAD_FAST	1 (elem)
	18	LOAD_FAST	0 (self)
	20	LOAD_ATTR	1 (start)
	22	COMPARE_OP	0 (<)
	24	POP_JUMP_IF_FALSE	30
477	26	LOAD_CONST	1 (False)
	28	RETURN_VALUE	

```

478      >> 30 LOAD_FAST          1 (elem)
           32 LOAD_FAST          0 (self)
           34 LOAD_ATTR          1 (start)
           36 BINARY_SUBTRACT
           38 LOAD_FAST          0 (self)
           40 LOAD_ATTR          0 (step)
           42 BINARY_MODULO
           44 LOAD_CONST         2 (0)
           46 COMPARE_OP         2 (==)
           48 RETURN_VALUE

```

Disassembly of <code object __getitem__ at 0x7f6cf32ac9d0, file "<dis>", line 479>

```

480      0 LOAD_GLOBAL          0 (_coconut)
         2 LOAD_METHOD          1 (isinstance)
         4 LOAD_FAST          1 (index)
         6 LOAD_GLOBAL          0 (_coconut)
         8 LOAD_ATTR          2 (slice)
        10 CALL_METHOD          2
        12 EXTENDED_ARG          1
        14 POP_JUMP_IF_FALSE    278
        16 LOAD_FAST          1 (index)
        18 LOAD_ATTR          3 (start)
        20 LOAD_CONST         0 (None)
        22 COMPARE_OP         8 (is)
        24 POP_JUMP_IF_TRUE     38
        26 LOAD_FAST          1 (index)
        28 LOAD_ATTR          3 (start)
        30 LOAD_CONST         1 (0)
        32 COMPARE_OP         5 (>=)
        34 EXTENDED_ARG          1
        36 POP_JUMP_IF_FALSE    278
      >> 38 LOAD_FAST          1 (index)
         40 LOAD_ATTR          4 (stop)
         42 LOAD_CONST         0 (None)
         44 COMPARE_OP         8 (is)
         46 POP_JUMP_IF_TRUE     60
         48 LOAD_FAST          1 (index)
        50 LOAD_ATTR          4 (stop)

```

		52 LOAD_CONST	1 (0)
		54 COMPARE_OP	5 (>=)
		56 EXTENDED_ARG	1
		58 POP_JUMP_IF_FALSE	278
481	>>	60 LOAD_FAST	0 (self)
		62 LOAD_ATTR	3 (start)
		64 LOAD_FAST	0 (self)
		66 LOAD_ATTR	5 (step)
		68 ROT_TWO	
		70 STORE_FAST	2 (new_start)
		72 STORE_FAST	3 (new_step)
482		74 LOAD_FAST	0 (self)
		76 LOAD_ATTR	5 (step)
		78 POP_JUMP_IF_FALSE	106
		80 LOAD_FAST	1 (index)
		82 LOAD_ATTR	3 (start)
		84 LOAD_CONST	0 (None)
		86 COMPARE_OP	9 (is not)
		88 POP_JUMP_IF_FALSE	106
483		90 LOAD_FAST	2 (new_start)
		92 LOAD_FAST	0 (self)
		94 LOAD_ATTR	5 (step)
		96 LOAD_FAST	1 (index)
		98 LOAD_ATTR	3 (start)
		100 BINARY_MULTIPLY	
		102 INPLACE_ADD	
		104 STORE_FAST	2 (new_start)
484	>>	106 LOAD_FAST	0 (self)
		108 LOAD_ATTR	5 (step)
		110 POP_JUMP_IF_FALSE	132
		112 LOAD_FAST	1 (index)
		114 LOAD_ATTR	5 (step)
		116 LOAD_CONST	0 (None)
		118 COMPARE_OP	9 (is not)

		120 POP_JUMP_IF_FALSE	132
485		122 LOAD_FAST	3 (new_step)
		124 LOAD_FAST	1 (index)
		126 LOAD_ATTR	5 (step)
		128 INPLACE_MULTIPLY	
		130 STORE_FAST	3 (new_step)
486	>>	132 LOAD_FAST	1 (index)
		134 LOAD_ATTR	4 (stop)
		136 LOAD_CONST	0 (None)
		138 COMPARE_OP	8 (is)
		140 POP_JUMP_IF_FALSE	154
487		142 LOAD_FAST	0 (self)
		144 LOAD_METHOD	6 (__class__)
		146 LOAD_FAST	2 (new_start)
		148 LOAD_FAST	3 (new_step)
		150 CALL_METHOD	2
		152 RETURN_VALUE	
488	>>	154 LOAD_FAST	0 (self)
		156 LOAD_ATTR	5 (step)
		158 POP_JUMP_IF_FALSE	220
		160 LOAD_GLOBAL	0 (_coconut)
		162 LOAD_METHOD	1 (isinstance)
		164 LOAD_FAST	0 (self)
		166 LOAD_ATTR	3 (start)
		168 LOAD_GLOBAL	0 (_coconut)
		170 LOAD_ATTR	7 (int)
		172 CALL_METHOD	2
		174 POP_JUMP_IF_FALSE	220
		176 LOAD_GLOBAL	0 (_coconut)
		178 LOAD_METHOD	1 (isinstance)
		180 LOAD_FAST	0 (self)
		182 LOAD_ATTR	5 (step)
		184 LOAD_GLOBAL	0 (_coconut)
		186 LOAD_ATTR	7 (int)

	188	CALL_METHOD	2
	190	POP_JUMP_IF_FALSE	220
489	192	LOAD_GLOBAL	0 (_coconut)
	194	LOAD_METHOD	8 (range)
	196	LOAD_FAST	2 (new_start)
	198	LOAD_FAST	0 (self)
	200	LOAD_ATTR	3 (start)
	202	LOAD_FAST	0 (self)
	204	LOAD_ATTR	5 (step)
	206	LOAD_FAST	1 (index)
	208	LOAD_ATTR	4 (stop)
	210	BINARY_MULTIPLY	
	212	BINARY_ADD	
	214	LOAD_FAST	3 (new_step)
	216	CALL_METHOD	3
	218	RETURN_VALUE	
490	>>	220 LOAD_GLOBAL	9 (_coconut_map)
		222 LOAD_FAST	0 (self)
		224 LOAD_ATTR	10 (__getitem__)
		226 LOAD_GLOBAL	0 (_coconut)
		228 LOAD_METHOD	8 (range)
		230 LOAD_FAST	1 (index)
		232 LOAD_ATTR	3 (start)
		234 LOAD_CONST	0 (None)
		236 COMPARE_OP	9 (is not)
		238 POP_JUMP_IF_FALSE	246
		240 LOAD_FAST	1 (index)
		242 LOAD_ATTR	3 (start)
		244 JUMP_FORWARD	2 (to 248)
	>>	246 LOAD_CONST	1 (0)
	>>	248 LOAD_FAST	1 (index)
		250 LOAD_ATTR	4 (stop)
		252 LOAD_FAST	1 (index)
		254 LOAD_ATTR	5 (step)
		256 LOAD_CONST	0 (None)
		258 COMPARE_OP	9 (is not)

	260	EXTENDED_ARG	1
	262	POP_JUMP_IF_FALSE	270
	264	LOAD_FAST	1 (index)
	266	LOAD_ATTR	5 (step)
	268	JUMP_FORWARD	2 (to 272)
>>	270	LOAD_CONST	2 (1)
>>	272	CALL_METHOD	3
	274	CALL_FUNCTION	2
	276	RETURN_VALUE	
491	>>	278 LOAD_FAST	1 (index)
		280 LOAD_CONST	1 (0)
		282 COMPARE_OP	0 (<)
		284 EXTENDED_ARG	1
		286 POP_JUMP_IF_FALSE	298
492		288 LOAD_GLOBAL	0 (_coconut)
		290 LOAD_METHOD	11 (IndexError)
		292 LOAD_CONST	3 ('count indices must be positive')
		294 CALL_METHOD	1
		296 RAISE_VARARGS	1
493	>>	298 LOAD_FAST	0 (self)
		300 LOAD_ATTR	5 (step)
		302 EXTENDED_ARG	1
		304 POP_JUMP_IF_FALSE	322
		306 LOAD_FAST	0 (self)
		308 LOAD_ATTR	3 (start)
		310 LOAD_FAST	0 (self)
		312 LOAD_ATTR	5 (step)
		314 LOAD_FAST	1 (index)
		316 BINARY_MULTIPLY	
		318 BINARY_ADD	
		320 RETURN_VALUE	
>>	322	LOAD_FAST	0 (self)
	324	LOAD_ATTR	3 (start)
	326	RETURN_VALUE	

Disassembly of <code object count at 0x7f6cf32aca80, file "<dis>", line 494>:

```

496          0 LOAD_FAST          0 (self)
            2 LOAD_ATTR          0 (step)
            4 POP_JUMP_IF_TRUE      30

497          6 LOAD_FAST          1 (elem)
            8 LOAD_FAST          0 (self)
           10 LOAD_ATTR          1 (start)
           12 COMPARE_OP          2 (==)
           14 POP_JUMP_IF_FALSE    26
           16 LOAD_GLOBAL        2 (_coconut)
           18 LOAD_METHOD        3 (float)
           20 LOAD_CONST         1 ('inf')
           22 CALL_METHOD        1
           24 RETURN_VALUE
>>          26 LOAD_CONST         2 (0)
           28 RETURN_VALUE

498    >>          30 LOAD_GLOBAL        4 (int)
            32 LOAD_FAST          1 (elem)
            34 LOAD_FAST          0 (self)
            36 COMPARE_OP          6 (in)
            38 CALL_FUNCTION      1
           40 RETURN_VALUE

```

Disassembly of <code object index at 0x7f6cf32acb30, file "<dis>", line 499>:

```

501          0 LOAD_FAST          1 (elem)
            2 LOAD_FAST          0 (self)
            4 COMPARE_OP          7 (not in)
            6 POP_JUMP_IF_FALSE    38

502          8 LOAD_GLOBAL        0 (_coconut)
           10 LOAD_METHOD        1 (ValueError)
           12 LOAD_GLOBAL        0 (_coconut)
           14 LOAD_METHOD        2 (repr)
           16 LOAD_FAST          1 (elem)
           18 CALL_METHOD        1
           20 LOAD_CONST         1 (' not in ')

```

```

22 BINARY_ADD
24 LOAD_GLOBAL          0 (_coconut)
26 LOAD_METHOD          2 (repr)
28 LOAD_FAST            0 (self)
30 CALL_METHOD          1
32 BINARY_ADD
34 CALL_METHOD          1
36 RAISE_VARARGS        1

503  >> 38 LOAD_FAST            0 (self)
      40 LOAD_ATTR              3 (step)
      42 POP_JUMP_IF_FALSE    60
      44 LOAD_FAST            1 (elem)
      46 LOAD_FAST            0 (self)
      48 LOAD_ATTR              4 (start)
      50 BINARY_SUBTRACT
      52 LOAD_FAST            0 (self)
      54 LOAD_ATTR              3 (step)
      56 BINARY_FLOOR_DIVIDE
      58 RETURN_VALUE
      >> 60 LOAD_CONST          2 (0)
      62 RETURN_VALUE

Disassembly of <code object __reversed__ at 0x7f6cf32acbe0, file "<dis>", line 504
505      0 LOAD_FAST            0 (self)
      2 LOAD_ATTR              0 (step)
      4 POP_JUMP_IF_TRUE     10

506      6 LOAD_FAST            0 (self)
      8 RETURN_VALUE

507  >> 10 LOAD_GLOBAL          1 (_coconut)
      12 LOAD_METHOD          2 (TypeError)
      14 LOAD_GLOBAL          3 (repr)
      16 LOAD_FAST            0 (self)
      18 CALL_FUNCTION        1
      20 LOAD_CONST          1 (' object is not reversible')
      22 BINARY_ADD

```

24	CALL_METHOD	1
26	RAISE_VARARGS	1
28	LOAD_CONST	0 (None)
30	RETURN_VALUE	

Disassembly of <code object __repr__ at 0x7f6cf32acc90, file "<dis>", line 508>:
509

0	LOAD_CONST	1 ('count(%r, %r)')
2	LOAD_FAST	0 (self)
4	LOAD_ATTR	0 (start)
6	LOAD_FAST	0 (self)
8	LOAD_ATTR	1 (step)
10	BUILD_TUPLE	2
12	BINARY_MODULO	
14	RETURN_VALUE	

Disassembly of <code object __hash__ at 0x7f6cf32acd40, file "<dis>", line 510>:
511

0	LOAD_GLOBAL	0 (_coconut)
2	LOAD_METHOD	1 (hash)
4	LOAD_FAST	0 (self)
6	LOAD_ATTR	2 (start)
8	LOAD_FAST	0 (self)
10	LOAD_ATTR	3 (step)
12	BUILD_TUPLE	2
14	CALL_METHOD	1
16	RETURN_VALUE	

Disassembly of <code object __reduce__ at 0x7f6cf32acdf0, file "<dis>", line 512>:
513

0	LOAD_FAST	0 (self)
2	LOAD_ATTR	0 (__class__)
4	LOAD_FAST	0 (self)
6	LOAD_ATTR	1 (start)
8	LOAD_FAST	0 (self)
10	LOAD_ATTR	2 (step)
12	BUILD_TUPLE	2
14	BUILD_TUPLE	2
16	RETURN_VALUE	

Disassembly of <code object __copy__ at 0x7f6cf32acea0, file "<dis>", line 514>:

```

515      0 LOAD_FAST      0 (self)
      2 LOAD_METHOD      0 (__class__)
      4 LOAD_FAST      0 (self)
      6 LOAD_ATTR        1 (start)
      8 LOAD_FAST      0 (self)
     10 LOAD_ATTR        2 (step)
     12 CALL_METHOD      2
     14 RETURN_VALUE

```

Disassembly of <code object __eq__ at 0x7f6cf32acf50, file "<dis>", line 516>:

```

517      0 LOAD_GLOBAL      0 (isinstance)
      2 LOAD_FAST      1 (other)
      4 LOAD_FAST      0 (self)
      6 LOAD_ATTR        1 (__class__)
      8 CALL_FUNCTION      2
     10 JUMP_IF_FALSE_OR_POP 34
     12 LOAD_FAST      0 (self)
     14 LOAD_ATTR        2 (start)
     16 LOAD_FAST      1 (other)
     18 LOAD_ATTR        2 (start)
     20 COMPARE_OP        2 (==)
     22 JUMP_IF_FALSE_OR_POP 34
     24 LOAD_FAST      0 (self)
     26 LOAD_ATTR        3 (step)
     28 LOAD_FAST      1 (other)
     30 LOAD_ATTR        3 (step)
     32 COMPARE_OP        2 (==)
    >> 34 RETURN_VALUE

```

Disassembly of <code object __fmap__ at 0x7f6cf32b1030, file "<dis>", line 518>:

```

519      0 LOAD_GLOBAL      0 (_coconut_map)
      2 LOAD_FAST      1 (func)
      4 LOAD_FAST      0 (self)
      6 CALL_FUNCTION      2
      8 RETURN_VALUE

```

Disassembly of <code object groupsof at 0x7f6cf32b1660, file "<dis>", line 520>:

```

520      0 LOAD_NAME        0 (__name__)

```

	2 STORE_NAME	1 (__module__)
	4 LOAD_CONST	0 ('groupsof')
	6 STORE_NAME	2 (__qualname__)
521	8 LOAD_CONST	1 ('groupsof(n, iterable) splits iterable
	10 STORE_NAME	3 (__doc__)
523	12 LOAD_CONST	2 (('group_size', 'iter'))
	14 STORE_NAME	4 (__slots__)
524	16 LOAD_CONST	3 (<code object __init__ at 0x7f6cf32b10e
	18 LOAD_CONST	4 ('groupsof.__init__')
	20 MAKE_FUNCTION	0
	22 STORE_NAME	5 (__init__)
532	24 LOAD_CONST	5 (<code object __iter__ at 0x7f6cf32b12f
	26 LOAD_CONST	6 ('groupsof.__iter__')
	28 MAKE_FUNCTION	0
	30 STORE_NAME	6 (__iter__)
545	32 LOAD_CONST	7 (<code object __len__ at 0x7f6cf32b1240
	34 LOAD_CONST	8 ('groupsof.__len__')
	36 MAKE_FUNCTION	0
	38 STORE_NAME	7 (__len__)
547	40 LOAD_CONST	9 (<code object __repr__ at 0x7f6cf32b13a
	42 LOAD_CONST	10 ('groupsof.__repr__')
	44 MAKE_FUNCTION	0
	46 STORE_NAME	8 (__repr__)
549	48 LOAD_CONST	11 (<code object __reduce__ at 0x7f6cf32b1
	50 LOAD_CONST	12 ('groupsof.__reduce__')
	52 MAKE_FUNCTION	0
	54 STORE_NAME	9 (__reduce__)
551	56 LOAD_CONST	13 (<code object __copy__ at 0x7f6cf32b150
	58 LOAD_CONST	14 ('groupsof.__copy__')
	60 MAKE_FUNCTION	0

	62	STORE_NAME	10	(__copy__)
553	64	LOAD_CONST	15	(<code object __fmap__ at 0x7f6cf32b15b
	66	LOAD_CONST	16	('groupsof.__fmap__')
	68	MAKE_FUNCTION	0	
	70	STORE_NAME	11	(__fmap__)
	72	LOAD_CONST	17	(None)
	74	RETURN_VALUE		

Disassembly of <code object __init__ at 0x7f6cf32b10e0, file "<dis>", line 524>:

525		0	LOAD_FAST	2	(iterable)
		2	LOAD_FAST	0	(self)
		4	STORE_ATTR	0	(iter)
526		6	SETUP_FINALLY	16	(to 24)
527		8	LOAD_GLOBAL	1	(_coconut)
		10	LOAD_METHOD	2	(int)
		12	LOAD_FAST	1	(n)
		14	CALL_METHOD	1	
		16	LOAD_FAST	0	(self)
		18	STORE_ATTR	3	(group_size)
		20	POP_BLOCK		
		22	JUMP_FORWARD	38	(to 62)
528	>>	24	DUP_TOP		
		26	LOAD_GLOBAL	1	(_coconut)
		28	LOAD_ATTR	4	(ValueError)
		30	COMPARE_OP	10	(exception match)
		32	POP_JUMP_IF_FALSE	60	
		34	POP_TOP		
		36	POP_TOP		
		38	POP_TOP		
529		40	LOAD_GLOBAL	1	(_coconut)
		42	LOAD_METHOD	5	(TypeError)
		44	LOAD_CONST	1	('group size must be an int; not %r')
		46	LOAD_FAST	1	(n)

```

        48 BUILD_TUPLE          1
        50 BINARY_MODULO
        52 CALL_METHOD          1
        54 RAISE_VARARGS        1
        56 POP_EXCEPT
        58 JUMP_FORWARD          2 (to 62)
>>    60 END_FINALLY

530    >>    62 LOAD_FAST          0 (self)
        64 LOAD_ATTR            3 (group_size)
        66 LOAD_CONST           2 (0)
        68 COMPARE_OP           1 (<=)
        70 POP_JUMP_IF_FALSE    90

531    72 LOAD_GLOBAL            1 (_coconut)
        74 LOAD_METHOD          4 (ValueError)
        76 LOAD_CONST           3 ('group size must be > 0; not %r')
        78 LOAD_FAST            0 (self)
        80 LOAD_ATTR            3 (group_size)
        82 BUILD_TUPLE          1
        84 BINARY_MODULO
        86 CALL_METHOD          1
        88 RAISE_VARARGS        1
>>    90 LOAD_CONST            0 (None)
        92 RETURN_VALUE

```

Disassembly of <code object __iter__ at 0x7f6cf32b12f0, file "<dis>", line 532>:

```

533    0 LOAD_GLOBAL            0 (_coconut)
        2 LOAD_METHOD          1 (iter)
        4 LOAD_FAST            0 (self)
        6 LOAD_ATTR            1 (iter)
        8 CALL_METHOD          1
       10 STORE_FAST            1 (iterator)

534    12 LOAD_CONST            1 (True)
       14 STORE_FAST            2 (loop)

535    >>    16 LOAD_FAST            2 (loop)

```

	18	POP_JUMP_IF_FALSE	114
536	20	BUILD_LIST	0
	22	STORE_FAST	3 (group)
537	24	LOAD_GLOBAL	0 (_coconut)
	26	LOAD_METHOD	2 (range)
	28	LOAD_FAST	0 (self)
	30	LOAD_ATTR	3 (group_size)
	32	CALL_METHOD	1
	34	GET_ITER	
>>	36	FOR_ITER	58 (to 96)
	38	STORE_FAST	4 (_)
538	40	SETUP_FINALLY	20 (to 62)
539	42	LOAD_FAST	3 (group)
	44	LOAD_METHOD	4 (append)
	46	LOAD_GLOBAL	0 (_coconut)
	48	LOAD_METHOD	5 (next)
	50	LOAD_FAST	1 (iterator)
	52	CALL_METHOD	1
	54	CALL_METHOD	1
	56	POP_TOP	
	58	POP_BLOCK	
	60	JUMP_ABSOLUTE	36
540	>>	62 DUP_TOP	
		64 LOAD_GLOBAL	0 (_coconut)
		66 LOAD_ATTR	6 (StopIteration)
		68 COMPARE_OP	10 (exception match)
		70 POP_JUMP_IF_FALSE	92
		72 POP_TOP	
		74 POP_TOP	
		76 POP_TOP	
541	78	LOAD_CONST	2 (False)
	80	STORE_FAST	2 (loop)


```

542      82 POP_EXCEPT
      84 POP_TOP
      86 JUMP_ABSOLUTE          96
      88 POP_EXCEPT
      90 JUMP_ABSOLUTE          36
    >>  92 END_FINALLY
      94 JUMP_ABSOLUTE          36

543    >>  96 LOAD_FAST          3 (group)
      98 POP_JUMP_IF_FALSE      16

544      100 LOAD_GLOBAL          0 (_coconut)
      102 LOAD_METHOD            7 (tuple)
      104 LOAD_FAST              3 (group)
      106 CALL_METHOD            1
      108 YIELD_VALUE
      110 POP_TOP
      112 JUMP_ABSOLUTE          16
    >>  114 LOAD_CONST          0 (None)
      116 RETURN_VALUE

```

Disassembly of <code object __len__ at 0x7f6cf32b1240, file "<dis>", line 545>:

```

546      0 LOAD_GLOBAL          0 (_coconut)
      2 LOAD_METHOD            1 (len)
      4 LOAD_FAST              0 (self)
      6 LOAD_ATTR              2 (iter)
      8 CALL_METHOD            1
     10 RETURN_VALUE

```

Disassembly of <code object __repr__ at 0x7f6cf32b13a0, file "<dis>", line 547>:

```

548      0 LOAD_CONST          1 ('groupsof(%r)')
      2 LOAD_FAST              0 (self)
      4 LOAD_ATTR              0 (iter)
      6 BUILD_TUPLE            1
      8 BINARY_MODULO
     10 RETURN_VALUE

```

Disassembly of <code object __reduce__ at 0x7f6cf32b1450, file "<dis>", line 549>:

```
550      0 LOAD_FAST      0 (self)
      2 LOAD_ATTR      0 (__class__)
      4 LOAD_FAST      0 (self)
      6 LOAD_ATTR      1 (group_size)
      8 LOAD_FAST      0 (self)
     10 LOAD_ATTR      2 (iter)
     12 BUILD_TUPLE      2
     14 BUILD_TUPLE      2
     16 RETURN_VALUE
```

Disassembly of <code object __copy__ at 0x7f6cf32b1500, file "<dis>", line 551>:

```
552      0 LOAD_FAST      0 (self)
      2 LOAD_METHOD      0 (__class__)
      4 LOAD_FAST      0 (self)
      6 LOAD_ATTR      1 (group_size)
      8 LOAD_GLOBAL      2 (_coconut)
     10 LOAD_ATTR      3 (copy)
     12 LOAD_METHOD      3 (copy)
     14 LOAD_FAST      0 (self)
     16 LOAD_ATTR      4 (iter)
     18 CALL_METHOD      1
     20 CALL_METHOD      2
     22 RETURN_VALUE
```

Disassembly of <code object __fmap__ at 0x7f6cf32b15b0, file "<dis>", line 553>:

```
554      0 LOAD_GLOBAL      0 (_coconut_map)
      2 LOAD_FAST      1 (func)
      4 LOAD_FAST      0 (self)
      6 CALL_FUNCTION      2
      8 RETURN_VALUE
```

Disassembly of <code object recursive_iterator at 0x7f6cf32b1a80, file "<dis>", line 555>:

```
555      0 LOAD_NAME      0 (__name__)
      2 STORE_NAME      1 (__module__)
      4 LOAD_CONST      0 ('recursive_iterator')
      6 STORE_NAME      2 (__qualname__)
```

556	8 LOAD_CONST 10 STORE_NAME	1 ('Decorator that optimizes a function f 3 (__doc__)
557	12 LOAD_CONST 14 STORE_NAME	2 (('func', 'tee_store', 'backup_tee_stor 4 (__slots__)
558	16 LOAD_CONST 18 LOAD_CONST 20 MAKE_FUNCTION 22 STORE_NAME	3 (<code object __init__ at 0x7f6cf32b171 4 ('recursive_iterator.__init__') 0 5 (__init__)
562	24 LOAD_CONST 26 LOAD_CONST 28 MAKE_FUNCTION 30 STORE_NAME	5 (<code object __call__ at 0x7f6cf32b170 6 ('recursive_iterator.__call__') 0 6 (__call__)
588	32 LOAD_CONST 34 LOAD_CONST 36 MAKE_FUNCTION 38 STORE_NAME	7 (<code object __repr__ at 0x7f6cf32b187 8 ('recursive_iterator.__repr__') 0 7 (__repr__)
590	40 LOAD_CONST 42 LOAD_CONST 44 MAKE_FUNCTION 46 STORE_NAME	9 (<code object __reduce__ at 0x7f6cf32b1 10 ('recursive_iterator.__reduce__') 0 8 (__reduce__)
592	48 LOAD_CONST 50 LOAD_CONST 52 LOAD_CONST 54 MAKE_FUNCTION 56 STORE_NAME 58 LOAD_CONST 60 RETURN_VALUE	14 ((None,)) 12 (<code object __get__ at 0x7f6cf32b19d0 13 ('recursive_iterator.__get__') 1 (defaults) 9 (__get__) 11 (None)

Disassembly of <code object __init__ at 0x7f6cf32b1710, file "<dis>", line 558>:

559	0 LOAD_FAST	1 (func)
	2 LOAD_FAST	0 (self)
	4 STORE_ATTR	0 (func)

560	6 BUILD_MAP	0
	8 LOAD_FAST	0 (self)
	10 STORE_ATTR	1 (tee_store)
561	12 BUILD_LIST	0
	14 LOAD_FAST	0 (self)
	16 STORE_ATTR	2 (backup_tee_store)
	18 LOAD_CONST	0 (None)
	20 RETURN_VALUE	

Disassembly of <code object __call__ at 0x7f6cf32b17c0, file "<dis>", line 562>:

563	0 LOAD_FAST	1 (args)
	2 LOAD_GLOBAL	0 (_coconut)
	4 LOAD_METHOD	1 (frozenset)
	6 LOAD_FAST	2 (kwargs)
	8 CALL_METHOD	1
	10 BUILD_TUPLE	2
	12 STORE_FAST	3 (key)
564	14 LOAD_CONST	1 (False)
	16 STORE_FAST	4 (use_backup)
565	18 SETUP_FINALLY	12 (to 32)
566	20 LOAD_GLOBAL	2 (hash)
	22 LOAD_FAST	3 (key)
	24 CALL_FUNCTION	1
	26 POP_TOP	
	28 POP_BLOCK	
	30 JUMP_FORWARD	68 (to 100)
567	>> 32 DUP_TOP	
	34 LOAD_GLOBAL	0 (_coconut)
	36 LOAD_ATTR	3 (Exception)
	38 COMPARE_OP	10 (exception match)
	40 POP_JUMP_IF_FALSE	98
	42 POP_TOP	

	44 POP_TOP	
	46 POP_TOP	
568	48 SETUP_FINALLY	18 (to 68)
569	50 LOAD_GLOBAL	0 (_coconut)
	52 LOAD_ATTR	4 (pickle)
	54 LOAD_METHOD	5 (dumps)
	56 LOAD_FAST	3 (key)
	58 LOAD_CONST	2 (-1)
	60 CALL_METHOD	2
	62 STORE_FAST	3 (key)
	64 POP_BLOCK	
	66 JUMP_FORWARD	26 (to 94)
570	>> 68 DUP_TOP	
	70 LOAD_GLOBAL	0 (_coconut)
	72 LOAD_ATTR	3 (Exception)
	74 COMPARE_OP	10 (exception match)
	76 POP_JUMP_IF_FALSE	92
	78 POP_TOP	
	80 POP_TOP	
	82 POP_TOP	
571	84 LOAD_CONST	3 (True)
	86 STORE_FAST	4 (use_backup)
	88 POP_EXCEPT	
	90 JUMP_FORWARD	2 (to 94)
	>> 92 END_FINALLY	
	>> 94 POP_EXCEPT	
	96 JUMP_FORWARD	2 (to 100)
	>> 98 END_FINALLY	
572	>> 100 LOAD_FAST	4 (use_backup)
	102 POP_JUMP_IF_FALSE	222
573	104 LOAD_GLOBAL	0 (_coconut)
	106 LOAD_METHOD	6 (enumerate)

	108	LOAD_FAST	0 (self)
	110	LOAD_ATTR	7 (backup_tee_store)
	112	CALL_METHOD	1
	114	GET_ITER	
>>	116	FOR_ITER	34 (to 152)
	118	UNPACK_SEQUENCE	2
	120	STORE_FAST	5 (i)
	122	UNPACK_SEQUENCE	2
	124	STORE_FAST	6 (k)
	126	STORE_FAST	7 (v)
574	128	LOAD_FAST	6 (k)
	130	LOAD_FAST	3 (key)
	132	COMPARE_OP	2 (==)
	134	POP_JUMP_IF_FALSE	116
575	136	LOAD_FAST	7 (v)
	138	LOAD_FAST	5 (i)
	140	ROT_TWO	
	142	STORE_FAST	8 (to_tee)
	144	STORE_FAST	9 (store_pos)
576	146	POP_TOP	
	148	JUMP_ABSOLUTE	168
	150	JUMP_ABSOLUTE	116
578	>>	152 LOAD_FAST	0 (self)
		154 LOAD_ATTR	8 (func)
		156 LOAD_FAST	1 (args)
		158 LOAD_FAST	2 (kwargs)
		160 CALL_FUNCTION_EX	1
		162 STORE_FAST	8 (to_tee)
579	164	LOAD_CONST	0 (None)
	166	STORE_FAST	9 (store_pos)
580	>>	168 LOAD_GLOBAL	9 (_coconut_tee)
		170 LOAD_FAST	8 (to_tee)

		172 CALL_FUNCTION	1
		174 UNPACK_SEQUENCE	2
		176 STORE_FAST	10 (to_store)
		178 STORE_FAST	11 (to_return)
581		180 LOAD_FAST	9 (store_pos)
		182 LOAD_CONST	0 (None)
		184 COMPARE_OP	8 (is)
		186 POP_JUMP_IF_FALSE	206
582		188 LOAD_FAST	0 (self)
		190 LOAD_ATTR	7 (backup_tee_store)
		192 LOAD_METHOD	10 (append)
		194 LOAD_FAST	3 (key)
		196 LOAD_FAST	10 (to_store)
		198 BUILD_LIST	2
		200 CALL_METHOD	1
		202 POP_TOP	
		204 JUMP_FORWARD	14 (to 220)
584	>>	206 LOAD_FAST	10 (to_store)
		208 LOAD_FAST	0 (self)
		210 LOAD_ATTR	7 (backup_tee_store)
		212 LOAD_FAST	9 (store_pos)
		214 BINARY_SUBSCR	
		216 LOAD_CONST	4 (1)
		218 STORE_SUBSCR	
	>>	220 JUMP_FORWARD	38 (to 260)
586	>>	222 LOAD_GLOBAL	9 (_coconut_tee)
		224 LOAD_FAST	0 (self)
		226 LOAD_ATTR	11 (tee_store)
		228 LOAD_METHOD	12 (get)
		230 LOAD_FAST	3 (key)
		232 CALL_METHOD	1
		234 JUMP_IF_TRUE_OR_POP	246
		236 LOAD_FAST	0 (self)
		238 LOAD_ATTR	8 (func)

```

240 LOAD_FAST          1 (args)
242 LOAD_FAST          2 (kwargs)
244 CALL_FUNCTION_EX   1
>> 246 CALL_FUNCTION   1
248 UNPACK_SEQUENCE    2
250 LOAD_FAST          0 (self)
252 LOAD_ATTR          11 (tee_store)
254 LOAD_FAST          3 (key)
256 STORE_SUBSCR
258 STORE_FAST         11 (to_return)

587 >> 260 LOAD_FAST         11 (to_return)
262 RETURN_VALUE

```

Disassembly of <code object __repr__ at 0x7f6cf32b1870, file "<dis>", line 588>:

```

589 0 LOAD_CONST         1 ('@recursive_iterator(')
2 LOAD_GLOBAL         0 (_coconut)
4 LOAD_METHOD         1 (repr)
6 LOAD_FAST          0 (self)
8 LOAD_ATTR          2 (func)
10 CALL_METHOD        1
12 BINARY_ADD
14 LOAD_CONST         2 (')')
16 BINARY_ADD
18 RETURN_VALUE

```

Disassembly of <code object __reduce__ at 0x7f6cf32b1920, file "<dis>", line 590>:

```

591 0 LOAD_FAST          0 (self)
2 LOAD_ATTR          0 (__class__)
4 LOAD_FAST          0 (self)
6 LOAD_ATTR          1 (func)
8 BUILD_TUPLE        1
10 BUILD_TUPLE        2
12 RETURN_VALUE

```

Disassembly of <code object __get__ at 0x7f6cf32b19d0, file "<dis>", line 592>:

```

593 0 LOAD_GLOBAL         0 (_coconut)
2 LOAD_ATTR          1 (functools)

```


	4	LOAD_METHOD	2	(partial)
	6	LOAD_FAST	0	(self)
	8	LOAD_FAST	1	(obj)
	10	CALL_METHOD	2	
	12	RETURN_VALUE		

Disassembly of <code object _coconut_FunctionMatchErrorContext at 0x7f6cf32b1df0,

594	0	LOAD_NAME	0	(__name__)
	2	STORE_NAME	1	(__module__)
	4	LOAD_CONST	0	('_coconut_FunctionMatchErrorContext')
	6	STORE_NAME	2	(__qualname__)
595	8	LOAD_CONST	1	(('exc_class', 'taken'))
	10	STORE_NAME	3	(__slots__)
596	12	LOAD_NAME	4	(_coconut)
	14	LOAD_ATTR	5	(threading)
	16	LOAD_METHOD	6	(local)
	18	CALL_METHOD	0	
	20	STORE_NAME	7	(threadlocal_var)
597	22	LOAD_CONST	2	(<code object __init__ at 0x7f6cf32b1b3
	24	LOAD_CONST	3	('_coconut_FunctionMatchErrorContext.')
	26	MAKE_FUNCTION	0	
	28	STORE_NAME	8	(__init__)
600	30	LOAD_CONST	4	(<code object __enter__ at 0x7f6cf32b1b
	32	LOAD_CONST	5	('_coconut_FunctionMatchErrorContext.')
	34	MAKE_FUNCTION	0	
	36	STORE_NAME	9	(__enter__)
605	38	LOAD_CONST	6	(<code object __exit__ at 0x7f6cf32b1c9
	40	LOAD_CONST	7	('_coconut_FunctionMatchErrorContext.')
	42	MAKE_FUNCTION	0	
	44	STORE_NAME	10	(__exit__)
607	46	LOAD_NAME	11	(classmethod)

```

608      48 LOAD_CONST      8 (<code object get at 0x7f6cf32b1d40, fi
      50 LOAD_CONST      9 ('_coconut_FunctionMatchErrorContext.ge
      52 MAKE_FUNCTION      0
      54 CALL_FUNCTION       1
      56 STORE_NAME        12 (get)
      58 LOAD_CONST       10 (None)
      60 RETURN_VALUE

```

Disassembly of <code object __init__ at 0x7f6cf32b1b30, file "<dis>", line 597>:

```

598      0 LOAD_FAST        1 (exc_class)
      2 LOAD_FAST        0 (self)
      4 STORE_ATTR        0 (exc_class)

```

```

599      6 LOAD_CONST        1 (False)
      8 LOAD_FAST        0 (self)
     10 STORE_ATTR        1 (taken)
     12 LOAD_CONST        0 (None)
     14 RETURN_VALUE

```

Disassembly of <code object __enter__ at 0x7f6cf32b1be0, file "<dis>", line 600>:

```

601      0 SETUP_FINALLY    18 (to 20)

602      2 LOAD_FAST        0 (self)
      4 LOAD_ATTR        0 (threadlocal_var)
      6 LOAD_ATTR        1 (contexts)
      8 LOAD_METHOD      2 (append)
     10 LOAD_FAST        0 (self)
     12 CALL_METHOD       1
     14 POP_TOP
     16 POP_BLOCK
     18 JUMP_FORWARD      32 (to 52)

```

```

603    >>  20 DUP_TOP
      22 LOAD_GLOBAL      3 (_coconut)
      24 LOAD_ATTR        4 (AttributeError)
      26 COMPARE_OP      10 (exception match)
      28 POP_JUMP_IF_FALSE 50
      30 POP_TOP

```

```

        32 POP_TOP
        34 POP_TOP

604      36 LOAD_FAST          0 (self)
        38 BUILD_LIST         1
        40 LOAD_FAST          0 (self)
        42 LOAD_ATTR           0 (threadlocal_var)
        44 STORE_ATTR         1 (contexts)
        46 POP_EXCEPT
        48 JUMP_FORWARD        2 (to 52)
>>     50 END_FINALLY
>>     52 LOAD_CONST          0 (None)
        54 RETURN_VALUE

```

Disassembly of <code object __exit__ at 0x7f6cf32b1c90, file "<dis>", line 605>:

```

606      0 LOAD_FAST          0 (self)
        2 LOAD_ATTR           0 (threadlocal_var)
        4 LOAD_ATTR           1 (contexts)
        6 LOAD_METHOD         2 (pop)
        8 CALL_METHOD         0
       10 POP_TOP
       12 LOAD_CONST          0 (None)
       14 RETURN_VALUE

```

Disassembly of <code object get at 0x7f6cf32b1d40, file "<dis>", line 607>:

```

609      0 SETUP_FINALLY       16 (to 18)

610      2 LOAD_FAST          0 (cls)
        4 LOAD_ATTR           0 (threadlocal_var)
        6 LOAD_ATTR           1 (contexts)
        8 LOAD_CONST         1 (-1)
       10 BINARY_SUBSCR
       12 STORE_FAST          1 (ctx)
       14 POP_BLOCK
       16 JUMP_FORWARD        32 (to 50)

611    >> 18 DUP_TOP
        20 LOAD_GLOBAL         2 (_coconut)

```

	22	LOAD_ATTR	3	(AttributeError)
	24	LOAD_GLOBAL	2	(_coconut)
	26	LOAD_ATTR	4	(IndexError)
	28	BUILD_TUPLE	2	
	30	COMPARE_OP	10	(exception match)
	32	POP_JUMP_IF_FALSE	48	
	34	POP_TOP		
	36	POP_TOP		
	38	POP_TOP		
612	40	LOAD_GLOBAL	5	(_coconut_MatchError)
	42	ROT_FOUR		
	44	POP_EXCEPT		
	46	RETURN_VALUE		
	>>	48	END_FINALLY	
613	>>	50	LOAD_FAST	1 (ctx)
		52	LOAD_ATTR	6 (taken)
		54	POP_JUMP_IF_TRUE	68
614		56	LOAD_CONST	2 (True)
		58	LOAD_FAST	1 (ctx)
		60	STORE_ATTR	6 (taken)
615		62	LOAD_FAST	1 (ctx)
		64	LOAD_ATTR	7 (exc_class)
		66	RETURN_VALUE	
616	>>	68	LOAD_GLOBAL	5 (_coconut_MatchError)
		70	RETURN_VALUE	

Disassembly of <code object _coconut_base_pattern_func at 0x7f6cf32c13a0, file "<

618	0	LOAD_NAME	0	(__name__)
	2	STORE_NAME	1	(__module__)
	4	LOAD_CONST	0	('_coconut_base_pattern_func')
	6	STORE_NAME	2	(__qualname__)
619	8	LOAD_CONST	1	((('FunctionMatchError', '__doc__', 'pat

	10 STORE_NAME	3 (__slots__)
620	12 LOAD_CONST	2 (True)
	14 STORE_NAME	4 (_coconut_is_match)
621	16 LOAD_CONST	3 (<code object __init__ at 0x7f6cf32b1ea
	18 LOAD_CONST	4 ('_coconut_base_pattern_func.__init__')
	20 MAKE_FUNCTION	0
	22 STORE_NAME	5 (__init__)
627	24 LOAD_CONST	5 (<code object add at 0x7f6cf32b1f50, fi
	26 LOAD_CONST	6 ('_coconut_base_pattern_func.add')
	28 MAKE_FUNCTION	0
	30 STORE_NAME	6 (add)
633	32 LOAD_CONST	7 (<code object __call__ at 0x7f6cf32c10e
	34 LOAD_CONST	8 ('_coconut_base_pattern_func.__call__')
	36 MAKE_FUNCTION	0
	38 STORE_NAME	7 (__call__)
641	40 LOAD_CONST	9 (<code object _coconut_tco_func at 0x7f
	42 LOAD_CONST	10 ('_coconut_base_pattern_func._coconut_t
	44 MAKE_FUNCTION	0
	46 STORE_NAME	8 (_coconut_tco_func)
649	48 LOAD_CONST	11 (<code object __repr__ at 0x7f6cf32c103
	50 LOAD_CONST	12 ('_coconut_base_pattern_func.__repr__')
	52 MAKE_FUNCTION	0
	54 STORE_NAME	9 (__repr__)
651	56 LOAD_CONST	13 (<code object __reduce__ at 0x7f6cf32c1
	58 LOAD_CONST	14 ('_coconut_base_pattern_func.__reduce__
	60 MAKE_FUNCTION	0
	62 STORE_NAME	10 (__reduce__)
653	64 LOAD_CONST	18 ((None,))
	66 LOAD_CONST	16 (<code object __get__ at 0x7f6cf32c12f0
	68 LOAD_CONST	17 ('_coconut_base_pattern_func.__get__')

70	MAKE_FUNCTION	1 (defaults)
72	STORE_NAME	11 (__get__)
74	LOAD_CONST	15 (None)
76	RETURN_VALUE	

Disassembly of <code object __init__ at 0x7f6cf32b1ea0, file "<dis>", line 621>:

622	0	LOAD_GLOBAL	0 (_coconut)
	2	LOAD_METHOD	1 (type)
	4	LOAD_GLOBAL	2 (_coconut_str)
	6	LOAD_CONST	1 ('MatchError')
	8	CALL_FUNCTION	1
	10	LOAD_GLOBAL	3 (_coconut_MatchError)
	12	BUILD_TUPLE	1
	14	BUILD_MAP	0
	16	CALL_METHOD	3
	18	LOAD_FAST	0 (self)
	20	STORE_ATTR	4 (FunctionMatchError)
623	22	LOAD_CONST	0 (None)
	24	LOAD_FAST	0 (self)
	26	STORE_ATTR	5 (__doc__)
624	28	BUILD_LIST	0
	30	LOAD_FAST	0 (self)
	32	STORE_ATTR	6 (patterns)
625	34	LOAD_FAST	1 (funcs)
	36	GET_ITER	
>>	38	FOR_ITER	14 (to 54)
	40	STORE_FAST	2 (func)
626	42	LOAD_FAST	0 (self)
	44	LOAD_METHOD	7 (add)
	46	LOAD_FAST	2 (func)
	48	CALL_METHOD	1
	50	POP_TOP	
	52	JUMP_ABSOLUTE	38
>>	54	LOAD_CONST	0 (None)

56 RETURN_VALUE

Disassembly of <code object add at 0x7f6cf32b1f50, file "<dis>", line 627>:

```

628      0 LOAD_GLOBAL           0 (_coconut)
        2 LOAD_METHOD             1 (getattr)
        4 LOAD_FAST               1 (func)
        6 LOAD_CONST            1 ('__doc__')
        8 LOAD_CONST            0 (None)
       10 CALL_METHOD             3
       12 JUMP_IF_TRUE_OR_POP    18
       14 LOAD_FAST              0 (self)
       16 LOAD_ATTR             2 (__doc__)
>>    18 LOAD_FAST              0 (self)
       20 STORE_ATTR            2 (__doc__)

629      22 LOAD_GLOBAL           0 (_coconut)
       24 LOAD_METHOD             3 (isinstance)
       26 LOAD_FAST               1 (func)
       28 LOAD_GLOBAL           4 (_coconut_base_pattern_func)
       30 CALL_METHOD             2
       32 POP_JUMP_IF_FALSE    52

630      34 LOAD_FAST              0 (self)
       36 DUP_TOP
       38 LOAD_ATTR             5 (patterns)
       40 LOAD_FAST               1 (func)
       42 LOAD_ATTR             5 (patterns)
       44 INPLACE_ADD
       46 ROT_TWO
       48 STORE_ATTR            5 (patterns)
       50 JUMP_FORWARD          12 (to 64)

632    >>    52 LOAD_FAST              0 (self)
        54 LOAD_ATTR             5 (patterns)
        56 LOAD_METHOD             6 (append)
        58 LOAD_FAST               1 (func)
        60 CALL_METHOD             1
        62 POP_TOP

```

```

>> 64 LOAD_CONST          0 (None)
    66 RETURN_VALUE

Disassembly of <code object __call__ at 0x7f6cf32c10e0, file "<dis>", line 633>:
634      0 LOAD_FAST        0 (self)
        2 LOAD_ATTR          0 (patterns)
        4 LOAD_CONST        0 (None)
        6 LOAD_CONST        1 (-1)
        8 BUILD_SLICE      2
       10 BINARY_SUBSCR
       12 GET_ITER
>>    14 FOR_ITER          78 (to 94)
        16 STORE_FAST       3 (func)

635      18 SETUP_FINALLY   50 (to 70)

636      20 LOAD_GLOBAL      1 (_coconut_FunctionMatchErrorContext)
        22 LOAD_FAST        0 (self)
        24 LOAD_ATTR          2 (FunctionMatchError)
        26 CALL_FUNCTION     1
        28 SETUP_WITH       30 (to 60)
        30 POP_TOP

637      32 LOAD_FAST        3 (func)
        34 LOAD_FAST        1 (args)
        36 LOAD_FAST        2 (kwargs)
        38 CALL_FUNCTION_EX  1
        40 POP_BLOCK
        42 ROT_TWO
        44 BEGIN_FINALLY
        46 WITH_CLEANUP_START
        48 WITH_CLEANUP_FINISH
        50 POP_FINALLY      0
        52 POP_BLOCK
        54 ROT_TWO
        56 POP_TOP
        58 RETURN_VALUE
>>    60 WITH_CLEANUP_START

```



```

        62 WITH_CLEANUP_FINISH
        64 END_FINALLY
        66 POP_BLOCK
        68 JUMP_ABSOLUTE          14

638    >> 70 DUP_TOP
        72 LOAD_FAST              0 (self)
        74 LOAD_ATTR              2 (FunctionMatchError)
        76 COMPARE_OP             10 (exception match)
        78 POP_JUMP_IF_FALSE      90
        80 POP_TOP
        82 POP_TOP
        84 POP_TOP

639    86 POP_EXCEPT
        88 JUMP_ABSOLUTE          14
    >> 90 END_FINALLY
        92 JUMP_ABSOLUTE          14

640    >> 94 LOAD_FAST              0 (self)
        96 LOAD_ATTR              0 (patterns)
        98 LOAD_CONST             1 (-1)
       100 BINARY_SUBSCR
       102 LOAD_FAST              1 (args)
       104 LOAD_FAST              2 (kwargs)
       106 CALL_FUNCTION_EX       1
       108 RETURN_VALUE

```

Disassembly of <code object _coconut_tco_func at 0x7f6cf32c1190, file "<dis>", line 642

```

642    0 LOAD_FAST              0 (self)
        2 LOAD_ATTR              0 (patterns)
        4 LOAD_CONST             0 (None)
        6 LOAD_CONST             1 (-1)
        8 BUILD_SLICE            2
       10 BINARY_SUBSCR
       12 GET_ITER
    >> 14 FOR_ITER                78 (to 94)
       16 STORE_FAST             3 (func)

```

643	18 SETUP_FINALLY	50 (to 70)
644	20 LOAD_GLOBAL	1 (_coconut_FunctionMatchErrorContext)
	22 LOAD_FAST	0 (self)
	24 LOAD_ATTR	2 (FunctionMatchError)
	26 CALL_FUNCTION	1
	28 SETUP_WITH	30 (to 60)
	30 POP_TOP	
645	32 LOAD_FAST	3 (func)
	34 LOAD_FAST	1 (args)
	36 LOAD_FAST	2 (kwargs)
	38 CALL_FUNCTION_EX	1
	40 POP_BLOCK	
	42 ROT_TWO	
	44 BEGIN_FINALLY	
	46 WITH_CLEANUP_START	
	48 WITH_CLEANUP_FINISH	
	50 POP_FINALLY	0
	52 POP_BLOCK	
	54 ROT_TWO	
	56 POP_TOP	
	58 RETURN_VALUE	
>>	60 WITH_CLEANUP_START	
	62 WITH_CLEANUP_FINISH	
	64 END_FINALLY	
	66 POP_BLOCK	
	68 JUMP_ABSOLUTE	14
646	>> 70 DUP_TOP	
	72 LOAD_FAST	0 (self)
	74 LOAD_ATTR	2 (FunctionMatchError)
	76 COMPARE_OP	10 (exception match)
	78 POP_JUMP_IF_FALSE	90
	80 POP_TOP	
	82 POP_TOP	
	84 POP_TOP	

```

647          86 POP_EXCEPT
          88 JUMP_ABSOLUTE          14
      >>    90 END_FINALLY
          92 JUMP_ABSOLUTE          14

648      >>    94 LOAD_GLOBAL          3 (_coconut_tail_call)
          96 LOAD_FAST          0 (self)
          98 LOAD_ATTR          0 (patterns)
         100 LOAD_CONST          1 (-1)
         102 BINARY_SUBSCR
         104 BUILD_TUPLE          1
         106 LOAD_FAST          1 (args)
         108 BUILD_TUPLE_UNPACK_WITH_CALL    2
         110 LOAD_FAST          2 (kwargs)
         112 CALL_FUNCTION_EX          1
         114 RETURN_VALUE

```

Disassembly of <code object __repr__ at 0x7f6cf32c1030, file "<dis>", line 649>:

```

650          0 LOAD_CONST          1 ('addpattern(')
          2 LOAD_GLOBAL          0 (_coconut)
          4 LOAD_METHOD          1 (repr)
          6 LOAD_FAST          0 (self)
          8 LOAD_ATTR          2 (patterns)
         10 LOAD_CONST          2 (0)
         12 BINARY_SUBSCR
         14 CALL_METHOD          1
         16 BINARY_ADD
         18 LOAD_CONST          3 (')(*)')
         20 BINARY_ADD
         22 LOAD_GLOBAL          0 (_coconut)
         24 LOAD_METHOD          1 (repr)
         26 LOAD_FAST          0 (self)
         28 LOAD_ATTR          2 (patterns)
         30 LOAD_CONST          4 (1)
         32 LOAD_CONST          0 (None)
         34 BUILD_SLICE          2
         36 BINARY_SUBSCR

```

```

38 CALL_METHOD          1
40 BINARY_ADD
42 LOAD_CONST           5 (')')
44 BINARY_ADD
46 RETURN_VALUE

```

Disassembly of <code object __reduce__ at 0x7f6cf32c1240, file "<dis>", line 651>:

```

652      0 LOAD_FAST          0 (self)
      2 LOAD_ATTR             0 (__class__)
      4 LOAD_GLOBAL           1 (_coconut)
      6 LOAD_METHOD          2 (tuple)
      8 LOAD_FAST          0 (self)
     10 LOAD_ATTR             3 (patterns)
     12 CALL_METHOD          1
     14 BUILD_TUPLE          2
     16 RETURN_VALUE

```

Disassembly of <code object __get__ at 0x7f6cf32c12f0, file "<dis>", line 653>:

```

654      0 LOAD_FAST          1 (obj)
      2 LOAD_CONST           0 (None)
      4 COMPARE_OP           8 (is)
      6 POP_JUMP_IF_FALSE   12

655      8 LOAD_FAST          0 (self)
     10 RETURN_VALUE

656  >>  12 LOAD_GLOBAL           0 (_coconut)
      14 LOAD_ATTR             1 (functools)
      16 LOAD_METHOD          2 (partial)
      18 LOAD_FAST          0 (self)
      20 LOAD_FAST          1 (obj)
      22 CALL_METHOD          2
      24 RETURN_VALUE

```

Disassembly of <code object _coconut_mark_as_match at 0x7f6cf32c1450, file "<dis>">:

```

658      0 LOAD_CONST           1 (True)
      2 LOAD_FAST          0 (base_func)
      4 STORE_ATTR           0 (_coconut_is_match)

```

659	6 LOAD_FAST	0 (base_func)
	8 RETURN_VALUE	

Disassembly of <code object addpattern at 0x7f6cf32c1500, file "<dis>", line 660>:

663	0 LOAD_FAST	1 (kwargs)
	2 LOAD_METHOD	0 (pop)
	4 LOAD_CONST	1 ('allow_any_func')
	6 LOAD_CONST	2 (False)
	8 CALL_METHOD	2
	10 STORE_FAST	2 (allow_any_func)

664	12 LOAD_FAST	2 (allow_any_func)
	14 POP_JUMP_IF_TRUE	60
	16 LOAD_GLOBAL	1 (_coconut)
	18 LOAD_METHOD	2 (getattr)
	20 LOAD_FAST	0 (base_func)
	22 LOAD_CONST	3 ('_coconut_is_match')
	24 LOAD_CONST	2 (False)
	26 CALL_METHOD	3
	28 POP_JUMP_IF_TRUE	60

665	30 LOAD_GLOBAL	1 (_coconut)
	32 LOAD_ATTR	3 (warnings)
	34 LOAD_ATTR	4 (warn)
	36 LOAD_CONST	4 ('Possible misuse of addpattern with no
	38 LOAD_GLOBAL	1 (_coconut)
	40 LOAD_METHOD	5 (repr)
	42 LOAD_FAST	0 (base_func)
	44 CALL_METHOD	1
	46 BINARY_ADD	
	48 LOAD_CONST	5 (' (pass allow_any_func=True to dismiss
	50 BINARY_ADD	
	52 LOAD_CONST	6 (2)
	54 LOAD_CONST	7 (('stacklevel',))
	56 CALL_FUNCTION_KW	2
	58 POP_TOP	

```

666     >> 60 LOAD_FAST          1 (kwargs)
          62 POP_JUMP_IF_FALSE      84

667         64 LOAD_GLOBAL        1 (_coconut)
          66 LOAD_METHOD          6 (TypeError)
          68 LOAD_CONST          8 ('addpattern() got unexpected keyword a
          70 LOAD_GLOBAL        1 (_coconut)
          72 LOAD_METHOD          5 (repr)
          74 LOAD_FAST          1 (kwargs)
          76 CALL_METHOD          1
          78 BINARY_ADD
          80 CALL_METHOD          1
          82 RAISE_VARARGS          1

668     >> 84 LOAD_GLOBAL        1 (_coconut)
          86 LOAD_ATTR            7 (functools)
          88 LOAD_METHOD          8 (partial)
          90 LOAD_GLOBAL        9 (_coconut_base_pattern_func)
          92 LOAD_FAST          0 (base_func)
          94 CALL_METHOD          2
          96 RETURN_VALUE

```

Disassembly of <code object _coconut_partial at 0x7f6cf32c1d40, file "<dis>", line

```

670         0 LOAD_NAME           0 (__name__)
          2 STORE_NAME            1 (__module__)
          4 LOAD_CONST          0 ('_coconut_partial')
          6 STORE_NAME            2 (__qualname__)

671         8 LOAD_CONST          1 (('func', '_argdict', '_arglen', '_star
          10 STORE_NAME            3 (__slots__)

672         12 LOAD_NAME          4 (hasattr)
          14 LOAD_NAME          5 (_coconut)
          16 LOAD_ATTR          6 (functools)
          18 LOAD_ATTR          7 (partial)
          20 LOAD_CONST          2 ('__doc__')
          22 CALL_FUNCTION          2
          24 POP_JUMP_IF_FALSE      36

```

673		26 LOAD_NAME	5 (_coconut)
		28 LOAD_ATTR	6 (functools)
		30 LOAD_ATTR	7 (partial)
		32 LOAD_ATTR	8 (__doc__)
		34 STORE_NAME	8 (__doc__)
674	>>	36 LOAD_CONST	3 (<code object __init__ at 0x7f6cf32c15b
		38 LOAD_CONST	4 ('_coconut_partial.__init__')
		40 MAKE_FUNCTION	0
		42 STORE_NAME	9 (__init__)
680		44 LOAD_CONST	5 (<code object __reduce__ at 0x7f6cf32c1
		46 LOAD_CONST	6 ('_coconut_partial.__reduce__')
		48 MAKE_FUNCTION	0
		50 STORE_NAME	10 (__reduce__)
682		52 LOAD_CONST	7 (<code object __setstate__ at 0x7f6cf32
		54 LOAD_CONST	8 ('_coconut_partial.__setstate__')
		56 MAKE_FUNCTION	0
		58 STORE_NAME	11 (__setstate__)
684		60 LOAD_NAME	12 (property)
685		62 LOAD_CONST	9 (<code object args at 0x7f6cf32c1870, f
		64 LOAD_CONST	10 ('_coconut_partial.args')
		66 MAKE_FUNCTION	0
		68 CALL_FUNCTION	1
		70 STORE_NAME	13 (args)
687		72 LOAD_CONST	11 (<code object __call__ at 0x7f6cf32c192
		74 LOAD_CONST	12 ('_coconut_partial.__call__')
		76 MAKE_FUNCTION	0
		78 STORE_NAME	14 (__call__)
702		80 LOAD_CONST	13 (<code object __repr__ at 0x7f6cf32c1a8
		82 LOAD_CONST	14 ('_coconut_partial.__repr__')
		84 MAKE_FUNCTION	0

86	STORE_NAME	15	(__repr__)
88	LOAD_CONST	15	(None)
90	RETURN_VALUE		

Disassembly of <code object __init__ at 0x7f6cf32c15b0, file "<dis>", line 674>:

675	0	LOAD_FAST	1	(func)
	2	LOAD_FAST	0	(self)
	4	STORE_ATTR	0	(func)
676	6	LOAD_FAST	2	(argdict)
	8	LOAD_FAST	0	(self)
	10	STORE_ATTR	1	(argdict)
677	12	LOAD_FAST	3	(arglen)
	14	LOAD_FAST	0	(self)
	16	STORE_ATTR	2	(arglen)
678	18	LOAD_FAST	4	(args)
	20	LOAD_FAST	0	(self)
	22	STORE_ATTR	3	(stargs)
679	24	LOAD_FAST	5	(kwargs)
	26	LOAD_FAST	0	(self)
	28	STORE_ATTR	4	(keywords)
	30	LOAD_CONST	0	(None)
	32	RETURN_VALUE		

Disassembly of <code object __reduce__ at 0x7f6cf32c1660, file "<dis>", line 680>:

681	0	LOAD_FAST	0	(self)
	2	LOAD_ATTR	0	(__class__)
	4	LOAD_FAST	0	(self)
	6	LOAD_ATTR	1	(func)
	8	LOAD_FAST	0	(self)
	10	LOAD_ATTR	2	(argdict)
	12	LOAD_FAST	0	(self)
	14	LOAD_ATTR	3	(arglen)
	16	BUILD_TUPLE	3	
	18	LOAD_FAST	0	(self)

20	LOAD_ATTR	4	(_stargs)
22	BINARY_ADD		
24	LOAD_FAST	0	(self)
26	LOAD_ATTR	5	(keywords)
28	BUILD_TUPLE	3	
30	RETURN_VALUE		

Disassembly of <code object __setstate__ at 0x7f6cf32c1710, file "<dis>", line 682

683	0	LOAD_FAST	1	(keywords)
	2	LOAD_FAST	0	(self)
	4	STORE_ATTR	0	(keywords)
	6	LOAD_CONST	0	(None)
	8	RETURN_VALUE		

Disassembly of <code object args at 0x7f6cf32c1870, file "<dis>", line 684>:

686	0	LOAD_GLOBAL	0	(_coconut)
	2	LOAD_METHOD	1	(tuple)
	4	LOAD_CLOSURE	0	(self)
	6	BUILD_TUPLE	1	
	8	LOAD_CONST	1	(<code object <genexpr> at 0x7f6cf32c1710, file "<dis>", line 686>)
	10	LOAD_CONST	2	(' _coconut_partial.args.<locals>.<genexpr>')
	12	MAKE_FUNCTION	8	(closure)
	14	LOAD_GLOBAL	0	(_coconut)
	16	LOAD_METHOD	2	(range)
	18	LOAD_DEREF	0	(self)
	20	LOAD_ATTR	3	(_arglen)
	22	CALL_METHOD	1	
	24	GET_ITER		
	26	CALL_FUNCTION	1	
	28	CALL_METHOD	1	
	30	LOAD_DEREF	0	(self)
	32	LOAD_ATTR	4	(_stargs)
	34	BINARY_ADD		
	36	RETURN_VALUE		

Disassembly of <code object <genexpr> at 0x7f6cf32c17c0, file "<dis>", line 686>:

686	0	LOAD_FAST	0	(.0)
>>	2	FOR_ITER	18	(to 22)

	4	STORE_FAST	1	(i)
	6	LOAD_DEREF	0	(self)
	8	LOAD_ATTR	0	(_argdict)
	10	LOAD_METHOD	1	(get)
	12	LOAD_FAST	1	(i)
	14	CALL_METHOD	1	
	16	YIELD_VALUE		
	18	POP_TOP		
	20	JUMP_ABSOLUTE	2	
>>	22	LOAD_CONST	0	(None)
	24	RETURN_VALUE		

Disassembly of <code object __call__ at 0x7f6cf32c1920, file "<dis>", line 687>:

688	0	BUILD_LIST	0	
	2	STORE_FAST	3	(callargs)
689	4	LOAD_CONST	1	(0)
	6	STORE_FAST	4	(argind)
690	8	LOAD_GLOBAL	0	(_coconut)
	10	LOAD_METHOD	1	(range)
	12	LOAD_FAST	0	(self)
	14	LOAD_ATTR	2	(_arglen)
	16	CALL_METHOD	1	
	18	GET_ITER		
>>	20	FOR_ITER	118	(to 140)
	22	STORE_FAST	5	(i)
691	24	LOAD_FAST	5	(i)
	26	LOAD_FAST	0	(self)
	28	LOAD_ATTR	3	(_argdict)
	30	COMPARE_OP	6	(in)
	32	POP_JUMP_IF_FALSE	52	
692	34	LOAD_FAST	3	(callargs)
	36	LOAD_METHOD	4	(append)
	38	LOAD_FAST	0	(self)
	40	LOAD_ATTR	3	(_argdict)

		42 LOAD_FAST	5 (i)
		44 BINARY_SUBSCR	
		46 CALL_METHOD	1
		48 POP_TOP	
		50 JUMP_ABSOLUTE	20
693	>>	52 LOAD_FAST	4 (argind)
		54 LOAD_GLOBAL	0 (_coconut)
		56 LOAD_METHOD	5 (len)
		58 LOAD_FAST	1 (args)
		60 CALL_METHOD	1
		62 COMPARE_OP	5 (>=)
		64 POP_JUMP_IF_FALSE	116
694		66 LOAD_GLOBAL	0 (_coconut)
		68 LOAD_METHOD	6 (TypeError)
		70 LOAD_CONST	2 ('expected at least ')
		72 LOAD_GLOBAL	0 (_coconut)
		74 LOAD_METHOD	7 (str)
		76 LOAD_FAST	0 (self)
		78 LOAD_ATTR	2 (_arglen)
		80 LOAD_GLOBAL	0 (_coconut)
		82 LOAD_METHOD	5 (len)
		84 LOAD_FAST	0 (self)
		86 LOAD_ATTR	3 (_argdict)
		88 CALL_METHOD	1
		90 BINARY_SUBTRACT	
		92 CALL_METHOD	1
		94 BINARY_ADD	
		96 LOAD_CONST	3 (' argument(s) to ')
		98 BINARY_ADD	
		100 LOAD_GLOBAL	0 (_coconut)
		102 LOAD_METHOD	8 (repr)
		104 LOAD_FAST	0 (self)
		106 CALL_METHOD	1
		108 BINARY_ADD	
		110 CALL_METHOD	1
		112 RAISE_VARARGS	1

		114 JUMP_ABSOLUTE	20
696	>>	116 LOAD_FAST	3 (callargs)
		118 LOAD_METHOD	4 (append)
		120 LOAD_FAST	1 (args)
		122 LOAD_FAST	4 (argind)
		124 BINARY_SUBSCR	
		126 CALL_METHOD	1
		128 POP_TOP	
697		130 LOAD_FAST	4 (argind)
		132 LOAD_CONST	4 (1)
		134 INPLACE_ADD	
		136 STORE_FAST	4 (argind)
		138 JUMP_ABSOLUTE	20
698	>>	140 LOAD_FAST	3 (callargs)
		142 LOAD_FAST	0 (self)
		144 LOAD_ATTR	9 (_stargs)
		146 INPLACE_ADD	
		148 STORE_FAST	3 (callargs)
699		150 LOAD_FAST	3 (callargs)
		152 LOAD_FAST	1 (args)
		154 LOAD_FAST	4 (argind)
		156 LOAD_CONST	0 (None)
		158 BUILD_SLICE	2
		160 BINARY_SUBSCR	
		162 INPLACE_ADD	
		164 STORE_FAST	3 (callargs)
700		166 LOAD_FAST	2 (kwargs)
		168 LOAD_METHOD	10 (update)
		170 LOAD_FAST	0 (self)
		172 LOAD_ATTR	11 (keywords)
		174 CALL_METHOD	1
		176 POP_TOP	

701	178	LOAD_FAST	0 (self)
	180	LOAD_ATTR	12 (func)
	182	LOAD_FAST	3 (callargs)
	184	LOAD_FAST	2 (kwargs)
	186	CALL_FUNCTION_EX	1
	188	RETURN_VALUE	

Disassembly of <code object __repr__ at 0x7f6cf32c1a80, file "<dis>", line 702>:

703	0	BUILD_LIST	0
	2	STORE_FAST	1 (args)
704	4	LOAD_GLOBAL	0 (_coconut)
	6	LOAD_METHOD	1 (range)
	8	LOAD_FAST	0 (self)
	10	LOAD_ATTR	2 (_arglen)
	12	CALL_METHOD	1
	14	GET_ITER	
>>	16	FOR_ITER	48 (to 66)
	18	STORE_FAST	2 (i)
705	20	LOAD_FAST	2 (i)
	22	LOAD_FAST	0 (self)
	24	LOAD_ATTR	3 (_argdict)
	26	COMPARE_OP	6 (in)
	28	POP_JUMP_IF_FALSE	54
706	30	LOAD_FAST	1 (args)
	32	LOAD_METHOD	4 (append)
	34	LOAD_GLOBAL	0 (_coconut)
	36	LOAD_METHOD	5 (repr)
	38	LOAD_FAST	0 (self)
	40	LOAD_ATTR	3 (_argdict)
	42	LOAD_FAST	2 (i)
	44	BINARY_SUBSCR	
	46	CALL_METHOD	1
	48	CALL_METHOD	1
	50	POP_TOP	
	52	JUMP_ABSOLUTE	16

708	>>	54 LOAD_FAST	1 (args)
		56 LOAD_METHOD	4 (append)
		58 LOAD_CONST	1 ('?')
		60 CALL_METHOD	1
		62 POP_TOP	
		64 JUMP_ABSOLUTE	16
709	>>	66 LOAD_FAST	0 (self)
		68 LOAD_ATTR	6 (_stargs)
		70 GET_ITER	
	>>	72 FOR_ITER	20 (to 94)
		74 STORE_FAST	3 (arg)
710		76 LOAD_FAST	1 (args)
		78 LOAD_METHOD	4 (append)
		80 LOAD_GLOBAL	0 (_coconut)
		82 LOAD_METHOD	5 (repr)
		84 LOAD_FAST	3 (arg)
		86 CALL_METHOD	1
		88 CALL_METHOD	1
		90 POP_TOP	
		92 JUMP_ABSOLUTE	72
711	>>	94 LOAD_GLOBAL	0 (_coconut)
		96 LOAD_METHOD	5 (repr)
		98 LOAD_FAST	0 (self)
		100 LOAD_ATTR	7 (func)
		102 CALL_METHOD	1
		104 LOAD_CONST	2 ('\$(')
		106 BINARY_ADD	
		108 LOAD_CONST	3 (', ')
		110 LOAD_METHOD	8 (join)
		112 LOAD_FAST	1 (args)
		114 CALL_METHOD	1
		116 BINARY_ADD	
		118 LOAD_CONST	4 (')')
		120 BINARY_ADD	

122 RETURN_VALUE

Disassembly of <code object consume at 0x7f6cf32c1df0, file "<dis>", line 712>:

```

714      0 LOAD_GLOBAL          0 (_coconut)
        2 LOAD_ATTR              1 (collections)
        4 LOAD_ATTR              2 (deque)
        6 LOAD_FAST            0 (iterable)
        8 LOAD_FAST            1 (keep_last)
       10 LOAD_CONST              1 (('maxlen',))
       12 CALL_FUNCTION_KW       2
       14 RETURN_VALUE

```

Disassembly of <code object starmap at 0x7f6cf32c25b0, file "<dis>", line 715>:

```

715      0 LOAD_NAME              0 (__name__)
        2 STORE_NAME              1 (__module__)
        4 LOAD_CONST              0 ('starmap')
        6 STORE_NAME            2 (__qualname__)

```

```

716      8 LOAD_CONST              1 (('func', 'iter'))
       10 STORE_NAME              3 (__slots__)

```

```

717     12 LOAD_NAME              4 (hasattr)
       14 LOAD_NAME              5 (_coconut)
       16 LOAD_ATTR              6 (itertools)
       18 LOAD_ATTR              7 (starmap)
       20 LOAD_CONST              2 ('__doc__')
       22 CALL_FUNCTION          2
       24 POP_JUMP_IF_FALSE      36

```

```

718     26 LOAD_NAME              5 (_coconut)
       28 LOAD_ATTR              6 (itertools)
       30 LOAD_ATTR              7 (starmap)
       32 LOAD_ATTR              8 (__doc__)
       34 STORE_NAME            8 (__doc__)

```

```

719    >> 36 LOAD_CONST          3 (<code object __new__ at 0x7f6cf32c1ea0>)
        38 LOAD_CONST          4 ('starmap.__new__')
        40 MAKE_FUNCTION        0

```

	42 STORE_NAME	9 (__new__)
724	44 LOAD_CONST	5 (<code object __getitem__ at 0x7f6cf32c20e0)
	46 LOAD_CONST	6 ('starmap.__getitem__')
	48 MAKE_FUNCTION	0
	50 STORE_NAME	10 (__getitem__)
728	52 LOAD_CONST	7 (<code object __reversed__ at 0x7f6cf32c20e0)
	54 LOAD_CONST	8 ('starmap.__reversed__')
	56 MAKE_FUNCTION	0
	58 STORE_NAME	11 (__reversed__)
730	60 LOAD_CONST	9 (<code object __len__ at 0x7f6cf32c20e0)
	62 LOAD_CONST	10 ('starmap.__len__')
	64 MAKE_FUNCTION	0
	66 STORE_NAME	12 (__len__)
732	68 LOAD_CONST	11 (<code object __repr__ at 0x7f6cf32c20e0)
	70 LOAD_CONST	12 ('starmap.__repr__')
	72 MAKE_FUNCTION	0
	74 STORE_NAME	13 (__repr__)
734	76 LOAD_CONST	13 (<code object __reduce__ at 0x7f6cf32c20e0)
	78 LOAD_CONST	14 ('starmap.__reduce__')
	80 MAKE_FUNCTION	0
	82 STORE_NAME	14 (__reduce__)
736	84 LOAD_CONST	15 (<code object __reduce_ex__ at 0x7f6cf32c20e0)
	86 LOAD_CONST	16 ('starmap.__reduce_ex__')
	88 MAKE_FUNCTION	0
	90 STORE_NAME	15 (__reduce_ex__)
738	92 LOAD_CONST	17 (<code object __copy__ at 0x7f6cf32c20e0)
	94 LOAD_CONST	18 ('starmap.__copy__')
	96 MAKE_FUNCTION	0
	98 STORE_NAME	16 (__copy__)
740	100 LOAD_CONST	19 (<code object __fmap__ at 0x7f6cf32c20e0)

102	LOAD_CONST	20 ('starmap.__fmap__')
104	MAKE_FUNCTION	0
106	STORE_NAME	17 (__fmap__)
108	LOAD_CONST	21 (None)
110	RETURN_VALUE	

Disassembly of <code object __new__ at 0x7f6cf32c1ea0, file "<dis>", line 719>:

720	0	LOAD_GLOBAL	0 (_coconut)
	2	LOAD_ATTR	1 (itertools)
	4	LOAD_ATTR	2 (starmap)
	6	LOAD_METHOD	3 (__new__)
	8	LOAD_FAST	0 (cls)
	10	LOAD_FAST	1 (function)
	12	LOAD_FAST	2 (iterable)
	14	CALL_METHOD	3
	16	STORE_FAST	3 (new_map)
721	18	LOAD_FAST	1 (function)
	20	LOAD_FAST	3 (new_map)
	22	STORE_ATTR	4 (func)
722	24	LOAD_FAST	2 (iterable)
	26	LOAD_FAST	3 (new_map)
	28	STORE_ATTR	5 (iter)
723	30	LOAD_FAST	3 (new_map)
	32	RETURN_VALUE	

Disassembly of <code object __getitem__ at 0x7f6cf32c1f50, file "<dis>", line 724>:

725	0	LOAD_GLOBAL	0 (_coconut)
	2	LOAD_METHOD	1 (isinstance)
	4	LOAD_FAST	1 (index)
	6	LOAD_GLOBAL	0 (_coconut)
	8	LOAD_ATTR	2 (slice)
	10	CALL_METHOD	2
	12	POP_JUMP_IF_FALSE	36
726	14	LOAD_FAST	0 (self)

```

16 LOAD_METHOD          3 (__class__)
18 LOAD_FAST            0 (self)
20 LOAD_ATTR            4 (func)
22 LOAD_GLOBAL          5 (_coconut_igetitem)
24 LOAD_FAST            0 (self)
26 LOAD_ATTR            6 (iter)
28 LOAD_FAST            1 (index)
30 CALL_FUNCTION         2
32 CALL_METHOD          2
34 RETURN_VALUE

727    >> 36 LOAD_FAST            0 (self)
        38 LOAD_ATTR            4 (func)
        40 LOAD_GLOBAL          5 (_coconut_igetitem)
        42 LOAD_FAST            0 (self)
        44 LOAD_ATTR            6 (iter)
        46 LOAD_FAST            1 (index)
        48 CALL_FUNCTION         2
        50 CALL_FUNCTION_EX      0
        52 RETURN_VALUE

```

Disassembly of <code object __reversed__ at 0x7f6cf32c2030, file "<dis>", line 728

```

729      0 LOAD_FAST            0 (self)
        2 LOAD_ATTR            0 (__class__)
        4 LOAD_FAST            0 (self)
        6 LOAD_ATTR            1 (func)
        8 BUILD_TUPLE         1
       10 LOAD_GLOBAL          2 (_coconut_reversed)
       12 LOAD_FAST            0 (self)
       14 LOAD_ATTR            3 (iter)
       16 CALL_FUNCTION         1
       18 BUILD_TUPLE_UNPACK_WITH_CALL    2
       20 CALL_FUNCTION_EX      0
       22 RETURN_VALUE

```

Disassembly of <code object __len__ at 0x7f6cf32c20e0, file "<dis>", line 730>:

```

731      0 LOAD_GLOBAL          0 (_coconut)
        2 LOAD_METHOD          1 (len)

```

4	LOAD_FAST	0 (self)
6	LOAD_ATTR	2 (iter)
8	CALL_METHOD	1
10	RETURN_VALUE	

Disassembly of <code object __repr__ at 0x7f6cf32c2190, file "<dis>", line 732>:
733

0	LOAD_CONST	1 ('starmap(%r, %r)')
2	LOAD_FAST	0 (self)
4	LOAD_ATTR	0 (func)
6	LOAD_FAST	0 (self)
8	LOAD_ATTR	1 (iter)
10	BUILD_TUPLE	2
12	BINARY_MODULO	
14	RETURN_VALUE	

Disassembly of <code object __reduce__ at 0x7f6cf32c2240, file "<dis>", line 734>:
735

0	LOAD_FAST	0 (self)
2	LOAD_ATTR	0 (__class__)
4	LOAD_FAST	0 (self)
6	LOAD_ATTR	1 (func)
8	LOAD_FAST	0 (self)
10	LOAD_ATTR	2 (iter)
12	BUILD_TUPLE	2
14	BUILD_TUPLE	2
16	RETURN_VALUE	

Disassembly of <code object __reduce_ex__ at 0x7f6cf32c22f0, file "<dis>", line 737>:
737

0	LOAD_FAST	0 (self)
2	LOAD_METHOD	0 (__reduce__)
4	CALL_METHOD	0
6	RETURN_VALUE	

Disassembly of <code object __copy__ at 0x7f6cf32c23a0, file "<dis>", line 738>:
739

0	LOAD_FAST	0 (self)
2	LOAD_METHOD	0 (__class__)
4	LOAD_FAST	0 (self)
6	LOAD_ATTR	1 (func)
8	LOAD_GLOBAL	2 (_coconut)

10	LOAD_ATTR	3 (copy)
12	LOAD_METHOD	3 (copy)
14	LOAD_FAST	0 (self)
16	LOAD_ATTR	4 (iter)
18	CALL_METHOD	1
20	CALL_METHOD	2
22	RETURN_VALUE	

Disassembly of <code object __fmap__ at 0x7f6cf32c2450, file "<dis>", line 740>:

741	0	LOAD_FAST	0 (self)
	2	LOAD_METHOD	0 (__class__)
	4	LOAD_GLOBAL	1 (_coconut_forward_compose)
	6	LOAD_FAST	0 (self)
	8	LOAD_ATTR	2 (func)
	10	LOAD_FAST	1 (func)
	12	CALL_FUNCTION	2
	14	LOAD_FAST	0 (self)
	16	LOAD_ATTR	3 (iter)
	18	CALL_METHOD	2
	20	RETURN_VALUE	

Disassembly of <code object makedata at 0x7f6cf32c2500, file "<dis>", line 742>:

744	0	LOAD_GLOBAL	0 (_coconut)
	2	LOAD_METHOD	1 (hasattr)
	4	LOAD_FAST	0 (data_type)
	6	LOAD_CONST	1 ('_make')
	8	CALL_METHOD	2
	10	POP_JUMP_IF_FALSE	36
	12	LOAD_GLOBAL	0 (_coconut)
	14	LOAD_METHOD	2 (issubclass)
	16	LOAD_FAST	0 (data_type)
	18	LOAD_GLOBAL	0 (_coconut)
	20	LOAD_ATTR	3 (tuple)
	22	CALL_METHOD	2
	24	POP_JUMP_IF_FALSE	36
745	26	LOAD_FAST	0 (data_type)
	28	LOAD_METHOD	4 (_make)

		30 LOAD_FAST	1 (args)
		32 CALL_METHOD	1
		34 RETURN_VALUE	
746	>>	36 LOAD_GLOBAL	0 (_coconut)
		38 LOAD_METHOD	2 (issubclass)
		40 LOAD_FAST	0 (data_type)
		42 LOAD_GLOBAL	0 (_coconut)
		44 LOAD_ATTR	5 (map)
		46 LOAD_GLOBAL	0 (_coconut)
		48 LOAD_ATTR	6 (range)
		50 LOAD_GLOBAL	0 (_coconut)
		52 LOAD_ATTR	7 (abc)
		54 LOAD_ATTR	8 (Iterator)
		56 BUILD_TUPLE	3
		58 CALL_METHOD	2
		60 POP_JUMP_IF_FALSE	66
747		62 LOAD_FAST	1 (args)
		64 RETURN_VALUE	
748	>>	66 LOAD_GLOBAL	0 (_coconut)
		68 LOAD_METHOD	2 (issubclass)
		70 LOAD_FAST	0 (data_type)
		72 LOAD_GLOBAL	0 (_coconut)
		74 LOAD_ATTR	9 (str)
		76 CALL_METHOD	2
		78 POP_JUMP_IF_FALSE	90
749		80 LOAD_CONST	2 (')
		82 LOAD_METHOD	10 (join)
		84 LOAD_FAST	1 (args)
		86 CALL_METHOD	1
		88 RETURN_VALUE	
750	>>	90 LOAD_FAST	0 (data_type)
		92 LOAD_FAST	1 (args)
		94 CALL_FUNCTION	1

96 RETURN_VALUE

Disassembly of <code object fmap at 0x7f6cf32c2710, file "<dis>", line 751>:

```

754      0 LOAD_GLOBAL          0 (_coconut)
        2 LOAD_METHOD              1 (hasattr)
        4 LOAD_FAST                1 (obj)
        6 LOAD_CONST              1 ('__fmap__')
        8 CALL_METHOD             2
       10 POP_JUMP_IF_FALSE      22

755      12 LOAD_FAST                1 (obj)
       14 LOAD_METHOD              2 (__fmap__)
       16 LOAD_FAST                0 (func)
       18 CALL_METHOD             1
       20 RETURN_VALUE

756  >>  22 LOAD_FAST                1 (obj)
        24 LOAD_ATTR               3 (__class__)
        26 LOAD_ATTR               4 (__module__)
        28 LOAD_CONST              2 ('numpy')
        30 COMPARE_OP             2 (==)
        32 POP_JUMP_IF_FALSE      58

757      34 LOAD_CONST              3 (0)
        36 LOAD_CONST              4 (('vectorize',))
        38 IMPORT_NAME              5 (numpy)
        40 IMPORT_FROM            6 (vectorize)
        42 STORE_FAST              2 (vectorize)
        44 POP_TOP

758      46 LOAD_FAST              2 (vectorize)
        48 LOAD_FAST              0 (func)
        50 CALL_FUNCTION          1
        52 LOAD_FAST              1 (obj)
        54 CALL_FUNCTION          1
        56 RETURN_VALUE

759  >>  58 LOAD_GLOBAL          7 (_coconut_makedata)

```

```

60 LOAD_FAST          1 (obj)
62 LOAD_ATTR          3 (__class__)
64 BUILD_TUPLE         1
66 LOAD_GLOBAL         0 (_coconut)
68 LOAD_METHOD         8 (isinstance)
70 LOAD_FAST          1 (obj)
72 LOAD_GLOBAL         0 (_coconut)
74 LOAD_ATTR          9 (abc)
76 LOAD_ATTR         10 (Mapping)
78 CALL_METHOD         2
80 POP_JUMP_IF_FALSE  96
82 LOAD_GLOBAL        11 (_coconut_starmap)
84 LOAD_FAST          0 (func)
86 LOAD_FAST          1 (obj)
88 LOAD_METHOD        12 (items)
90 CALL_METHOD         0
92 CALL_FUNCTION       2
94 JUMP_FORWARD        8 (to 104)
>> 96 LOAD_GLOBAL        13 (_coconut_map)
98 LOAD_FAST          0 (func)
100 LOAD_FAST          1 (obj)
102 CALL_FUNCTION       2
>> 104 BUILD_TUPLE_UNPACK_WITH_CALL    2
106 CALL_FUNCTION_EX    0
108 RETURN_VALUE

```

Disassembly of <code object memoize at 0x7f6cf32c2660, file "<dis>", line 760>:

```

763      0 LOAD_GLOBAL      0 (_coconut)
      2 LOAD_ATTR           1 (functools)
      4 LOAD_ATTR           2 (lru_cache)
      6 LOAD_FAST          0 (maxsize)
      8 BUILD_TUPLE         1
     10 LOAD_FAST           1 (args)
     12 BUILD_TUPLE_UNPACK_WITH_CALL    2
     14 LOAD_FAST           2 (kwargs)
     16 CALL_FUNCTION_EX    1
     18 RETURN_VALUE

```

```

Disassembly of <code object sieve at 0x7f6cf32c2c90, file "<dis>", line 768>:
771          0 LOAD_CONST          1 (False)
            2 STORE_FAST          2 (_coconut_match_check)

772          4 LOAD_GLOBAL          0 (_coconut_get_function_match_error)
            6 CALL_FUNCTION          0
            8 STORE_FAST          3 (_coconut_FunctionMatchError)

773         10 LOAD_GLOBAL          1 (_coconut)
            12 LOAD_METHOD          2 (len)
            14 LOAD_FAST           0 (_coconut_match_to_args)
            16 CALL_METHOD          1
            18 LOAD_CONST          2 (1)
            20 COMPARE_OP          2 (==)
            22 POP_JUMP_IF_FALSE    112
            24 LOAD_GLOBAL          1 (_coconut)
            26 LOAD_METHOD          3 (isinstance)
            28 LOAD_FAST           0 (_coconut_match_to_args)
            30 LOAD_CONST          3 (0)
            32 BINARY_SUBSCR
            34 LOAD_GLOBAL          1 (_coconut)
            36 LOAD_ATTR            4 (abc)
            38 LOAD_ATTR            5 (Iterable)
            40 CALL_METHOD          2
            42 POP_JUMP_IF_FALSE    112

774         44 LOAD_GLOBAL          1 (_coconut)
            46 LOAD_METHOD          6 (iter)
            48 LOAD_FAST           0 (_coconut_match_to_args)
            50 LOAD_CONST          3 (0)
            52 BINARY_SUBSCR
            54 CALL_METHOD          1
            56 STORE_DEREF          0 (numbers)

775         58 LOAD_GLOBAL          1 (_coconut)
            60 LOAD_METHOD          7 (tuple)
            62 LOAD_GLOBAL          8 (_coconut_igetitem)
            64 LOAD_DEREF          0 (numbers)

```


	66	LOAD_GLOBAL	1	(_coconut)	
	68	LOAD_METHOD	9	(slice)	
	70	LOAD_CONST	0	(None)	
	72	LOAD_CONST	2	(1)	
	74	CALL_METHOD	2		
	76	CALL_FUNCTION	2		
	78	CALL_METHOD	1		
	80	STORE_FAST	4	(_coconut_match_temp_0)	
776	82	LOAD_GLOBAL	1	(_coconut)	
	84	LOAD_METHOD	2	(len)	
	86	LOAD_FAST	4	(_coconut_match_temp_0)	
	88	CALL_METHOD	1		
	90	LOAD_CONST	2	(1)	
	92	COMPARE_OP	2	(==)	
	94	POP_JUMP_IF_FALSE	112		
	96	LOAD_FAST	1	(_coconut_match_to_kwargs)	
	98	POP_JUMP_IF_TRUE	112		
777	100	LOAD_FAST	4	(_coconut_match_temp_0)	
	102	LOAD_CONST	3	(0)	
	104	BINARY_SUBSCR			
	106	STORE_DEREF	1	(p)	
778	108	LOAD_CONST	4	(True)	
	110	STORE_FAST	2	(_coconut_match_check)	
779	>>	112	LOAD_FAST	2	(_coconut_match_check)
		114	POP_JUMP_IF_TRUE	184	
780	116	LOAD_GLOBAL	1	(_coconut)	
	118	LOAD_METHOD	10	(repr)	
	120	LOAD_FAST	0	(_coconut_match_to_args)	
	122	CALL_METHOD	1		
	124	STORE_FAST	5	(_coconut_match_val_repr)	
781	126	LOAD_FAST	3	(_coconut_FunctionMatchError)	
	128	LOAD_CONST	5	("pattern-matching failed for 'def sieve"	

	130	LOAD_GLOBAL	1	(_coconut)	
	132	LOAD_METHOD	2	(len)	
	134	LOAD_FAST	5	(_coconut_match_val_repr)	
	136	CALL_METHOD	1		
	138	LOAD_CONST	6	(500)	
	140	COMPARE_OP	1	(<=)	
	142	POP_JUMP_IF_FALSE	148		
	144	LOAD_FAST	5	(_coconut_match_val_repr)	
	146	JUMP_FORWARD	14	(to 162)	
>>	148	LOAD_FAST	5	(_coconut_match_val_repr)	
	150	LOAD_CONST	0	(None)	
	152	LOAD_CONST	6	(500)	
	154	BUILD_SLICE	2		
	156	BINARY_SUBSCR			
	158	LOAD_CONST	7	('...')	
	160	BINARY_ADD			
>>	162	BINARY_ADD			
	164	CALL_FUNCTION	1		
	166	STORE_FAST	6	(_coconut_match_err)	
782	168	LOAD_CONST	8	('def sieve((p) :: numbers) =')	
	170	LOAD_FAST	6	(_coconut_match_err)	
	172	STORE_ATTR	11	(pattern)	
783	174	LOAD_FAST	0	(_coconut_match_to_args)	
	176	LOAD_FAST	6	(_coconut_match_err)	
	178	STORE_ATTR	12	(value)	
784	180	LOAD_FAST	6	(_coconut_match_err)	
	182	RAISE_VARARGS	1		
786	>>	184	LOAD_GLOBAL	13	(_coconut_tail_call)
		186	LOAD_GLOBAL	1	(_coconut)
		188	LOAD_ATTR	14	(itertools)
		190	LOAD_ATTR	15	(chain)
		192	LOAD_ATTR	16	(from_iterable)
		194	LOAD_CONST	9	(<code object <genexpr> at 0x7f6cf32c27
		196	LOAD_CONST	10	('sieve.<locals>.<genexpr>')

198	MAKE_FUNCTION	0
200	LOAD_CLOSURE	1 (p)
202	BUILD_TUPLE	1
204	LOAD_CONST	11 (<code object <lambda> at 0x7f6cf32c2920)
206	LOAD_CONST	12 ('sieve.<locals>.<lambda>')
208	MAKE_FUNCTION	8 (closure)
210	LOAD_CLOSURE	0 (numbers)
212	LOAD_CLOSURE	1 (p)
214	BUILD_TUPLE	2
216	LOAD_CONST	13 (<code object <lambda> at 0x7f6cf32c2a80)
218	LOAD_CONST	12 ('sieve.<locals>.<lambda>')
220	MAKE_FUNCTION	8 (closure)
222	BUILD_TUPLE	2
224	GET_ITER	
226	CALL_FUNCTION	1
228	CALL_FUNCTION	2
230	RETURN_VALUE	

Disassembly of <code object <genexpr> at 0x7f6cf32c27c0, file "<dis>", line 786>:

786	0	LOAD_FAST	0 (.0)
>>	2	FOR_ITER	12 (to 16)
	4	STORE_FAST	1 (_coconut_func)
	6	LOAD_FAST	1 (_coconut_func)
	8	CALL_FUNCTION	0
	10	YIELD_VALUE	
	12	POP_TOP	
	14	JUMP_ABSOLUTE	2
>>	16	LOAD_CONST	0 (None)
	18	RETURN_VALUE	

Disassembly of <code object <lambda> at 0x7f6cf32c2920, file "<dis>", line 786>:

786	0	LOAD_CONST	1 (<code object <genexpr> at 0x7f6cf32c27c0)
	2	LOAD_CONST	2 ('sieve.<locals>.<lambda>.<locals>.<genexpr>')
	4	MAKE_FUNCTION	0
	6	LOAD_CLOSURE	0 (p)
	8	BUILD_TUPLE	1
	10	LOAD_CONST	3 (<code object <lambda> at 0x7f6cf32c2870)
	12	LOAD_CONST	4 ('sieve.<locals>.<lambda>.<locals>.<lambda>')

14	MAKE_FUNCTION	8 (closure)
16	BUILD_TUPLE	1
18	GET_ITER	
20	CALL_FUNCTION	1
22	RETURN_VALUE	

Disassembly of <code object <genexpr> at 0x7f6cf32c27c0, file "<dis>", line 786>:

786	0	LOAD_FAST	0 (.0)
>>	2	FOR_ITER	12 (to 16)
	4	STORE_FAST	1 (_coconut_func)
	6	LOAD_FAST	1 (_coconut_func)
	8	CALL_FUNCTION	0
	10	YIELD_VALUE	
	12	POP_TOP	
	14	JUMP_ABSOLUTE	2
>>	16	LOAD_CONST	0 (None)
	18	RETURN_VALUE	

Disassembly of <code object <lambda> at 0x7f6cf32c2870, file "<dis>", line 786>:

786	0	LOAD_DEREF	0 (p)
	2	RETURN_VALUE	

Disassembly of <code object <lambda> at 0x7f6cf32c2a80, file "<dis>", line 786>:

786	0	LOAD_GLOBAL	0 (sieve)
	2	LOAD_GLOBAL	1 (filter)
	4	LOAD_CLOSURE	1 (p)
	6	BUILD_TUPLE	1
	8	LOAD_CONST	1 (<code object <lambda> at 0x7f6cf32c29d0)
	10	LOAD_CONST	2 ('sieve.<locals>.<lambda>.<locals>.<lambda>')
	12	MAKE_FUNCTION	8 (closure)
	14	LOAD_DEREF	0 (numbers)
	16	CALL_FUNCTION	2
	18	CALL_FUNCTION	1
	20	RETURN_VALUE	

Disassembly of <code object <lambda> at 0x7f6cf32c29d0, file "<dis>", line 786>:

786	0	LOAD_FAST	0 (n)
	2	LOAD_DEREF	0 (p)

4	BINARY_MODULO	
6	LOAD_CONST	1 (0)
8	COMPARE_OP	3 (!=)
10	RETURN_VALUE	

Disassembly of <code object <lambda> at 0x7f6cf32c2d40, file "<dis>", line 789>:

789	0	LOAD_FAST	0 (p)
	2	LOAD_CONST	1 (100)
	4	COMPARE_OP	0 (<)
	6	RETURN_VALUE	