```python
def inplace_quick_sort(S, a, b):
  """Sort the list from S[a] to S[b] inclusive using the quick-sort
algorithm."""
  if a >= b: return                                       # range is trivially
sorted
  pivot = S[b]                                            # last element of range
is pivot
  left = a                                                # will scan rightward
  right = b-1                                             # will scan leftward
  while left <= right:
    # scan until reaching value equal or larger than pivot (or right marker)
    while left <= right and S[left] < pivot:
      left += 1
    # scan until reaching value equal or smaller than pivot (or left marker)
    while left <= right and pivot < S[right]:
      right -= 1
    if left <= right:                                     # scans did not
strictly cross
      S[left], S[right] = S[right], S[left]               # swap values
      left, right = left + 1, right - 1                   # shrink range

  # put pivot into its final place (currently marked by left index)

  S[left], S[b] = S[b], S[left]

  # make recursive calls

  inplace_quick_sort(S, a, left - 1)
  inplace_quick_sort(S, left + 1, b)
```