SELECT(A, p, r, i)

```
1   while (r − p + 1) mod 5 ≠ 0
2       for j = p + 1 to r                    // put the minimum into A[p]
3           if A[p] > A[j]
4               exchange A[p] with A[j]
5       // If we want the minimum of A[p : r], we're done.
6       if i == 1
7           return A[p]
8       // Otherwise, we want the (i − 1)st element of A[p + 1 : r].
9       p = p + 1
10      i = i − 1
11  g = (r − p + 1)/5                          // number of 5-element groups
12  for j = p to p + g − 1                     // sort each group
13      sort ⟨A[j], A[j + g], A[j + 2g], A[j + 3g], A[j + 4g]⟩ in place
14  // All group medians now lie in the middle fifth of A[p : r].
15  // Find the pivot x recursively as the median of the group medians.
16  x = SELECT(A, p + 2g, p + 3g − 1, ⌈g/2⌉)
17  q = PARTITION-AROUND(A, p, r, x)          // partition around the pivot
18  // The rest is just like lines 3–9 of RANDOMIZED-SELECT.
19  k = q − p + 1
20  if i == k
21      return A[q]                            // the pivot value is the answer
22  elseif i < k
23      return SELECT(A, p, q − 1, i)
24  else return SELECT(A, q + 1, r, i − k)
```