Class work: Some-Sort-2

Note: As usual, we denote the size of A by n.

```
Some-Sort-2(A)
1  for k = 1 to ?
2      j = the index of the largest element among A[0], A[1], ..., A[n − k]
3      swap(A, j, n − k)
```

1. Write pseudocode for line 2 in the algorithm.
2. What can you say about A after one execution of the outer loop?
3. What is the case after two executions of the outer loop?
4. What should the last value of k be?
5. Now argue that that algorithm is correct by arguing that after the outer loop finishes executing, the input is always sorted.
6. Show how this works on $A = (3, 1, 5, 7, 4, 6, 2)$ by showing A after each execution of the outer loop.
7. Can you think of any ways to improve this code? If so, are they worth it?