More details about the moves made by back-and-forth Quicksort

The pivot value, A[1], is copied into x, which is one move, and then later into some position q, which is another move. And then the value that was at q must go somewhere, which is a third move. So there are at least these three moves. [Actually, if q=1 then there are only two such moves, since it is x itself that was at q to start, and it moves out and back in, so q=1 is a special case.]

In addition (when q>1) there are the possible moves made by the n-2 elements A[2], ..., A[q-1], A[q+1], ..., A[n]. These n-2 elements only move if they change sides, from left of q to the right, or vice versa. There are a total of n-1 positions each of them can end up in at the end of a run of BFPartition: q-1 positions on the left of q, and n-q on the right. (They cannot end up at position q.) These q-1 on the left will be filled by the q-1 smallest elements in the array, and the n-q on the right by the n-q largest.

Thus any one of these elements on the left will move to the right side of q, iff it is one of the n-q largest elements, ie if it is either the (q+1)st, or the (q+2)nd, ..., or the nth smallest in size. Each of these is equally likely. In fact the probability that any particular element e (of the n-2 elements whose moves we still need to account for) is the i-th smallest in the array (for i=1,...q-1,q+1,...,n) is simply 1/(n-1). [The chance that e is the q-th smallest is 0.]

So the chance that element e ends up to the right is just the probability that it is the (q+1)-st smallest or the (q+2)-nd smallest, ... , or the n-th smallest, ie 1/(n-1) + ... + 1/(n-1) = (n-q)/(n-1).

Similarly, the probability of e ending up somewhere on the left is just that of its being either the smallest element, of the 2nd, ... , or the (q-1)-st, ie, (q-2)/(n-1).

Finally, the expected number that move from the left to the right is simply the number there are on the left (q-2) times the probability of moving to the right: (q-2)(n-q)/(n-1). And the expected number that move from right to left is the number on the right (n-q) times the probability of moving left: (n-q)(q-1)/(n-1).

Finally, the expected number of moves (for q>1) is 3 plus the two above expressions: 3 + (q-2)(n-q)/(n-1) + (n-q)(q-1)/(n-1).

Recall that if q=1 then there are only 2 moves. But letting q be one in the expression we got when Q>1, we see we *still* get 2, so we can use 3 + (q-2)(n-q)/(n-1) + (n-q)(q-1)/(n-1) for all cases.