**A** low-key sorting problem as [Le Monde current mathematical puzzle](https://xianblog.wordpress.com/2011/09/03/le-monde-puzzle-website/):

*If the numbers from 1 to 67 are randomly permuted and if the sorting algorithm consists in picking a number i with a position higher than its rank i and moving it at the correct i-th position, what is the maximal number of steps to sort this set of integers when the selected integer is chosen optimaly?*

As the question does not clearly state what happens to the number j that stood in the i-th position, I made the assumption that the entire sequence from position i to position n is moved one position upwards (rather than having i and j exchanged). In which case my intuition was that moving the smallest moveable number was optimal, resulting in the following R code

sorite<-function(permu){ n=length(permu) p=0 while(max(abs(permu-(1:n)))>0){

j=min(permu[permu<(1:n)])

p=p+1

permu=unique(c(permu[(1:n)

which takes at most n-1 steps to reorder the sequence. I however checked this insertion sort was indeed the case through a recursive function

resorite<-function(permu){ n=length(permu);p=0 while(max(abs(permu-(1:n)))>0){

j=cand=permu[permu<(1:n)]

if (length(cand)==1){

p=p+1

permu=unique(c(permu[(1:n)

which did confirm my intuition.