1. Assume we are sorting n distinct values. We say that the input is *random* if each permutation is equally likely. Donald Trump (who happens be taking this course in his spare time) suggests an alternative definition: An input is random if, for each pair of elements x, y, half the time x is before y (so that half the time y is before x). It is clear that if each permutation is equally likely then the list is random in the sense of Donald Trump.

   We might wonder if the converse holds. In other words, if the list is random in the sense of Donald Trump, is each permutation equally likely? It is plausable that the converse holds for some values of n and not others. For what values of n does the converse hold? Justify your answer.

2. Consider an insertion-sort-like algorithm. for sorting a list L with an even number of elements n: Assume that the first $i-2$ elements are sorted (where i is even). Compare the next two elements: $L[i-1]$ and $L[i]$. Put the larger element into its proper location in the sorted array using linear search (as in standard insertion sort). Then, starting from where the larger element ended up, put the smaller element into its proper location using linear search.

   (a) Write the pseudo code, using a sentinel, for this algorithm. To keep things consistent, just sort the first two elements ($L[1]$ and $L[2]$), and then start the outside iteration with elements ($L[3]$ and $L[4]$).

   (b) Assume $n = 4$. What is the best-case number of comparisons? Just state the number and show your input. Otherwise, no justification needed.

   (c) Assume $n = 4$. What is the worst-case number of comparisons? Just state the number and show your input. Otherwise, no justification needed.

   (d) Assume $n = 4$. What is the average-case number of comparisons? Justify. HINT: There are 24 possible orderings, which is a lot. It becomes more manageable if you realize that the first pair of elements essentially defines the whole ordering.

   (e) Calculate the number of comparisons the algorithm uses in the average case (for n even). Show your work.

   (f) How does the number of comparisons compare with the average case of insertion sort?