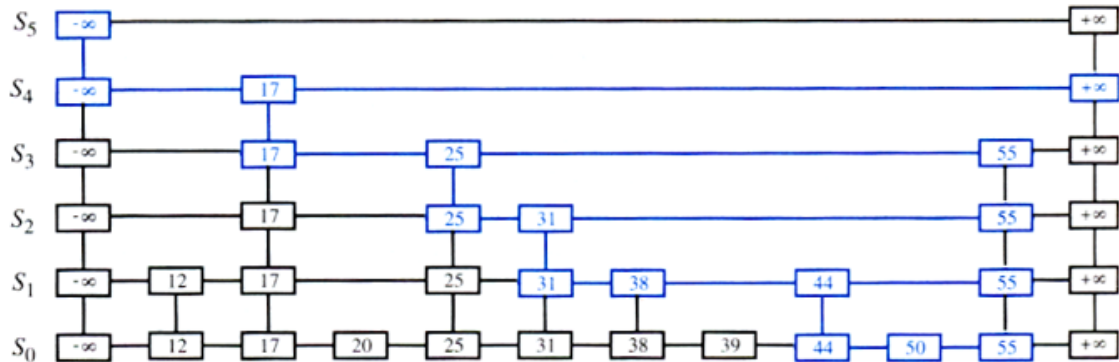# 3805ICT Advanced Algorithms
Randomised Algorithms Problems

Skip Lists

1. Write the C++ code for searching a skip list. Test it by tracing the path to find 50 in the skip list below.

2. Write C++ code that, starting with an empty skip list, performs the insert function. Test your code by showing the resulting skip list when the following keys are inserted with the shown "randomly generated" values. Draw the state of the skip list after each insertion:

   - 5, 1
   - 26, 1
   - 25, 4
   - 6, 3
   - 21, 1
   - 3, 2
   - 22, 2

3. Write C++ code that, starting with the skip list created in (b) above, performs the delete function in the order listed above.

Treaps

1. Treaps can be made self-adjusting by:

   - using the number of accesses as a priority and performing rotations as needed after each access.

   - Generating a random number each time an item is A is accessed. If this is smaller than A's current priority then use it as A's new priority (performing the appropriate rotations).

   Compare these methods with the randomised strategy.

2. Given a sequence of additions / deletions to a treap. Does the order in which these are done change the final structure of the treap?