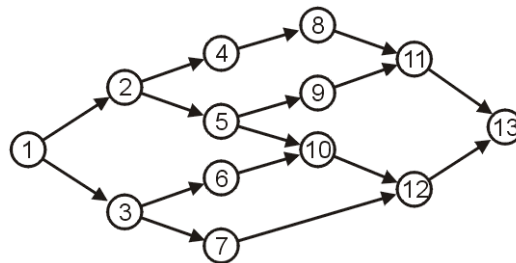


ECE-250 – Algorithms and Data Structures (Winter 2012)
Additional (exam-like) problems (2012-03-17)

1 – Run the first two iterations of quick sort, using the median-of-tree approximation for the median/pivot, on the following sequence: 18, 12, 35, 9, 27, 29, 5, 21, 10, 17.

2 – A simpler technique for finding the pivot is picking an arbitrary value, such as the first one or the last one. Show, for each of these two cases, a “hand crafted” sequence of values that causes quick sort to achieve its worst-case run time (i.e., a sequence of values that makes it run in $\Theta(n^2)$)

3 – Show the execution of topological sort, using an array of in-degrees and a queue, for the DAG shown below



4 – A Hamiltonian cycle in a graph is a path that visits each vertex exactly once and ends in the starting vertex. Suppose that we have some “magic oracle” (a hypothetical procedure) that determines if a given graph has a Hamiltonian cycle (presumably it does it efficiently).

Show that we can use this procedure to construct a Hamiltonian cycle in a graph, provided that it has one. In other words, show that finding a Hamiltonian cycle in a graph reduces to determining whether a graph has a Hamiltonian cycle.

Notice: this one is a non-trivial question—something like this could be an exam question, but most likely in the bonus marks category, if anything. However, you’ll probably agree with me that this is one of those problems that *after* you see (or think of) the solution, the problem (or rather, the solution) will seem obvious.