1. (15 points) Give an $O(VE)$-time algorithm for computing the transitive closure of a directed graph $G = (V, E)$. Compute its asymptotic running time.

2. (15 points) Grog –master of pictures– needs your help to compute the in- and out-degrees of all vertices in a directed multigraph G. However, he is not sure how to represent the graph so that the calculation is most efficient. For each of the three possible representations, express your answers in asymptotic notation (the only notation Grog understands), in terms of V and E, and justify your claim.

   (a) An *adjacency matrix* representation. Assume the size of the matrix is known.

   (b) An *edge list* representation. Assume vertices have arbitrary labels.

   (c) An *adjacency list* representation. Assume the vector's length is known.

3. (40 points) Consider a valleyed array $A[1, 2, \ldots, n]$ with the property that the subarray $A[1 \ldots i]$ has the property that $A[j] > A[j+1]$ for $1 \le j < i$, and the subarray $A[i \ldots n]$ has the property that $A[j] < A[j+1]$ for $i \le j < n$. For example, $A = [16, 15, 10, 9, 7, 3, 6, 8, 17, 23]$ is a valleyed array.

   (a) Write a recursive algorithm that takes asymptotically sub-linear time to find the minimum element of A.

   (b) Prove that your algorithm is correct. (Hint: prove that your algorithm's correctness follows from the correctness of another correct algorithm we already know.)

   (c) Now consider the multi-valleyed generalization, in which the array contains k valleys, i.e., it contains k subarrays, each of which is itself a valleyed array. Let $k = 2$ and prove that your algorithm can fail on such an input.

   (d) Suppose that $k = 2$ and we can guarantee that neither valley is closer than $n = 4$ positions to the middle of the array, and that the "joining point" of the two singly valleyed subarrays lays in the middle half of the array. Now write an algorithm that returns the minimum element of A in sublinear time. Prove that your algorithm is correct, give a recurrence relation for its running time, and solve for its asymptotic behavior.