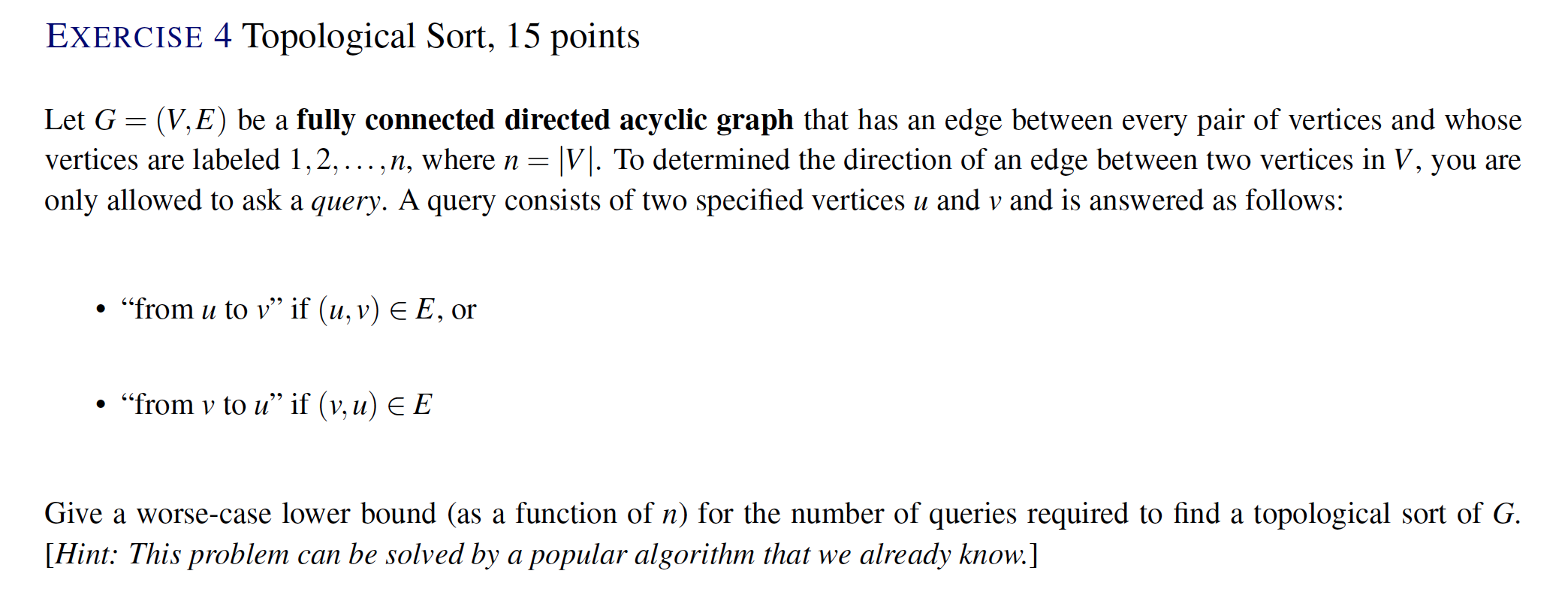
**Assignment 3 Exercise 4**



A query can be perceived as a comparison operator between two vertices u and v. The ordering of u and v can be determined as the output of a query (u,v) which has possible outcomes u > v or v > u. Because the graph is fully connected, there is an ordering between every combination of vertices u and v. Because the graph is directed and acyclic, an ordered list of vertices with a beginning and end can be derived. This DAG model is analogous to a simple list of numbers, which can be sorted into an ordered list using any comparison sort, i.e. quick sort. The “largest” vertex would be a source with directed edges going off to all other nodes. The “smallest” vertex would be a sink with directed edges coming in from all other nodes. Since the query is a “comparison” operator, we can use any comparison sort (i.e. quick sort) to generate a topological sort of G which is lower bounded by .

Therefore, there is a tight lower bound for the number of queries to perform a topological sort on G.