Computational Classes of Problems

For each of these problems, or languages, give its best known computational class. For example, the answer could be \( P \), \( \text{NP} \), \( \text{NP} \)-complete, \( \text{P}\text{-SPACE} \), recursive, recursively enumerable, to name just a few. For example, if a problem is known to be in the class \( \text{NP} \), but is not known to be in \( P \), and is also not known to be \( \text{NP} \)-complete, your answer would be “\( \text{NP} \)”.

If there is no class with a standard definition which contains the problem, you can say, “Not a member of any class that I can find.” That could be the correct answer!

1. Given a graph \( G \), is \( G \) planar? (That is, can it be drawn in a plane with no crossings?)

2. Given a room and various pieces of furniture and equipment, it is possible for those items to fit into the room?

3. Given a room with a door, and various pieces of furniture and equipment, is it possible to move those items into the room through the door? (This is not the same question!)

4. Does a context-free grammar generate all string? More specifically, given a context-free grammar \( G \) where \( \Sigma \) is the set of terminals of \( G \), is it true that \( L(G) = \Sigma^* \)?

5. Given an \( n \times n \) checkerboard, for some \( n \), and given a configuration of checkers on that board, can the black player win?

6. Given a Turing machine \( M \) and a number \( t \), will \( M \) halt within \( t \) steps?

7. Does a given general grammar \( G \) generate a given string \( w \)?

8. Given a set of jobs and a set of workers, where each worker is trained to work some given subset of the jobs, each job takes a given amount of time, and pairs of jobs \( (X,Y) \) are given, where \( X \) must be finished before work on \( Y \) begins, can all the jobs be finished within \( T \) hours?

9. We define a partial inversion of a string to be the string obtained reversing any substring. For example, \text{abaa}cd\text{ab} is a partial inversion of \text{abad}ca\text{ab}. Given strings \( u \) and \( v \) and a number \( k \), is it possible to obtain \( v \) from \( u \) by a sequence of \( k \) partial inversions?