1. (50 points) For each of the following problems, answer True, False, or Open Question. Do not assume any unproven hypotheses. Do not give reasons for your answers.

   _______PATH ∈ P
   _______PATH is NP-complete
   _______PATH ≤ₚ CLIQUE
   _______PATH ≤ₚ PATH
   _______PATH ∈ DSPACE(1)
   _______HAM-PATH ∈ P
   _______HAM-PATH ∈ NP
   _______HAM-PATH ∈ PSPACE
   _______HAM-PATH ≤ₚ PATH
   _______HAM-PATH is NP-complete
   _______The language PRIMES₁ = {1ᵏ | k is prime} is in P.
   _______P ⊆ NP ∩ co-NP
   _______If M is a DFA and M accepts all strings in language L, then L is regular.

2. (20 points) Consider the following scheduling problem. There is a list of final exams $F_1, \ldots, F_k$ to be scheduled. There is also a list of students $S_1, \ldots, S_m$. Each student is taking some specified subset of these exams. We would like to schedule these exams into time slots so that no student is required to take two exams in the same slot. The problem is to determine if there exists a schedule using only $h$ slots.

   We can formulate this as a language as follows:

   $\text{FINAL} = \{<S_1, \ldots, S_m, F_1, \ldots, F_k, T_1, \ldots, T_m, h> | T_i ⊆ \{F_i\} \text{ and there is a valid schedule using no more than } h \text{ slots}\}$.

   In this definition, $T_i$ is the set of final exams that student $i$ must take.

   Show that this language is NP-complete. (Hint: reduce from the 3-COLOR problem from HW #7, recalling that 3-COLOR = \{<G> | the nodes of G can be colored with 3 colors such that no two nodes joined by an edge have the same color\}.)

3. (30 points) Recall from HW #4 that a \textit{linearly bounded automaton} (LBA) is a deterministic 1-tape Turing machine whose head is not permitted to move off the input portion of the tape.

   Recall also that $K_{LBA} = \{<M, w> | M \text{ is an LBA accepting } w\}$.

   (a) Sketch an algorithm showing that TQBF is decidable with an LBA.

   (b) Show that $K_{LBA}$ is PSPACE-complete.
Solutions to Example Exam #2

1. \( T \text{PATH} \in P \)
   \( O \text{PATH} \) is NP-complete
   \( T \text{PATH} \leq_p \text{CLIQUE} \)
   \( T \text{PATH} \leq_p \text{PATH} \)
   \( F \text{PATH} \in \text{DSPACE}(1) \)
   \( O \text{HAM-PATH} \in P \)
   \( T \text{HAM-PATH} \in \text{NP} \)
   \( T \text{HAM-PATH} \in \text{PSPACE} \)
   \( O \text{HAM-PATH} \leq_p \text{PATH} \)
   \( T \text{HAM-PATH} \) is NP-complete
   \( T \) The language PRIMES\(_1\) = \{1^k \mid k \text{ is prime}\} is in P.
   \( T \) \( P \subseteq \text{NP} \cap \text{co-NP} \)
   \( F \) If M is a DFA and M accepts all strings in language L, then L is regular.

2. \( \text{FINAL} \in \text{NP} \) because, given a schedule, we can easily check to ensure that no student is required to take 2 exams in the same slot by checking each set \( T_i \).

   Now we will show that 3-COLOR \( \leq_p \text{FINAL} \). For input G to 3-COLOR, let each vertex correspond to an exam. Assume each student must take 2 exams, and let each edge correspond to a student and his/her two assigned exams. Let \( h \) equal 3. This is clearly polynomial. Now we show that this reduction works.

   \( \Rightarrow \) If G \( \in \text{3-COLOR} \), then we can produce a valid schedule. The color of the vertex (i.e., exam) corresponds to the time the exam is scheduled. Since no two vertices connected by an edge can be colored the same, the two exams that each student must take must be scheduled for different times.

   \( \Leftarrow \) If there is a valid schedule in \( h = 3 \) time slots, we can produce a 3-COLORing of G. For all exams in the same time slot, color the corresponding vertices the same. Pick a different color for each time slot. This must be a 3-COLORing, because each edge corresponds to a student’s two assigned exams (which correspond to 2 neighboring vertices), which cannot be scheduled for the same time (i.e., colored the same) if this is a valid schedule.

3. (a) An LBA is equivalent to a TM running in \( O(n) \) space. Create the LBA A to decide TQBF as follows:

   \( A = \text{“On input } \phi, \text{"} \)
   \( \bullet \) If \( \phi = \exists x \Psi \) then recursively call A on \( \Psi \) with 0, then 1 substituted for x. Accept if either accept.
   \( \bullet \) If \( \phi = \forall x \Psi \) then recursively call A on \( \Psi \) with 0, then 1 substituted for x. Accept if both accept.
   \( \bullet \) If \( \phi \) has no quantifiers, then evaluate and accept if true"

   The amount of space needed is that for storing the recursion stack, which is as deep as the number of quantifiers. So, this gives \( O(n) \) space.

   (b) Show that TQBF \( \leq_p K_{LBA} \) (which is sufficient, since TQBF is PSPACE-complete.) The reduction function \( f \) is as follows:

   \( f(\phi) = <A, \phi>, \) where A is the LBA from part 3(a) above. Clearly, \( \phi \in \text{TQBF iff } <A, \phi> \in K_{LBA} \).