Problem 1.

(a) Carefully explain why NP $\subseteq$ PSPACE.

(b) Suppose that A is PSPACE-hard. Explain why it follows that A is NP-hard. (To say A is PSPACE-hard means that any language in PSPACE can be reduced to A in polynomial time. We do not require that A, itself, is in PSPACE. Similarly, we say A is NP-hard if every language in NP has a polynomial-time reduction to A.)

(c) Suppose that every NP-hard language is PSPACE-hard. Would that imply NP = PSPACE? Explain.

Problem 2. Let A and B be languages, and say $A \leq_{\text{PSPACE}} B$ if there is function $f\colon \Sigma^* \to \Sigma^*$ that is computable by a TM using polynomial *space* and such that $w \in A$ if and only if $f(w) \in B$. We call f a *polynomial space reduction* from A to B. In the lecture for today, Sipser states that it would be a terrible idea to use $\leq_{\text{PSPACE}}$ in place of polynomial *time* reductions to define PSPACE completeness. To see why, let B be any language in PSPACE such that $B \neq \emptyset$ and $B \neq \Sigma^*$. Given any language A in PSPACE, give a polynomial space reduction f from A to B.

Problem 3. The Japanese game go-moku is played by two players, X and O, on a 19×19 grid. Players take turns placing stones on the intersection points in the grid, and the first player to achieve five of their stones consecutively in a row, column, or diagonal is the winner. Consider this game generalized to an $n \times n$ board. Let

$$GM = \{\langle P \rangle : P \text{ is a position in generalized go-moku,} \\ \text{where player } X \text{ has a winning strategy}\}.$$

By a *position* P we mean a board with stones placed on it, such as may occur in the middle of a play of the game, together with an indication of which player moves next. Show that $GM \in$ PSPACE by describing a TM with numbered steps that accepts GM. Besides creating the TM, explain why it runs in polynomial space.

Problem 4. (bonus) The cat-and-mouse game is played by two players, "Cat" and "Mouse", on an arbitrary undirected graph. At a given point, each player occupies a node of the graph. The players take turns moving to a node adjacent to the one that they currently occupy. A special node of the graph is called "Hole". Cat wins if the two players ever occupy the same node. Mouse wins if it reaches the Hole before the preceding happens. The game is a draw if a situation repeats (i.e., the two players simultaneously occupy positions that they simultaneously occupied previously, and it is the same player's turn to move). Define

$$\text{HAPPY-CAT} = \{\langle G, c, m, h \rangle \mid G, c, m, h \text{ are respectively a graph, and} \\ \text{positions of the Cat, Mouse, and Hole, such that} \\ \text{Cat has a winning strategy if Cat moves first}\}.$$

Show that HAPPY-CAT is in P. (Hint: The solution is not complicated and doesn't depend on subtle details in the way the game is defined. Consider the entire game tree. It is exponentially big, but you can search it in polynomial time.)