The more challenging problems are marked with $*$. Be concise when describing a Turing machine. It is like writing pseudocodes. It suffices to present the most important ideas behind your Turing machines. You do not need to give all the details, e.g., the set of states and the transition function. Check the course website for more info about homeworks. CC: Computational Complexity. TC: Introduction to the Theory of Computation. MA: Computational Complexity: A Modern Approach.

1. Exercises 7.7 on page 271 and Problem 7.14 on page 272 of TC (6 points): Show that NP is closed under union, concatenation, and star operation. (Xi: You may want to use the alternative view of NP as languages decided by polynomial-time verifiers. Can you also show that P is closed under the star operation?) Exercise 7.11 on page 272 of TC (4 points): Call graphs G and H isomorphic if the nodes of G may be reordered so that it is identical to H. Let

$$\text{ISO} = \big\{(G, H) \mid G \text{ and } H \text{ are isomorphic graphs}\big\}.$$

Show that ISO $\in$ NP.

2. Problem 8.20 on page 304 of TC (10 points): An undirected graph is bipartite if its nodes may be divided into two sets so that all edges go from a node in one set to a node in the other set. Show that a graph is bipartite if and only if it does't contain a cycle that has an odd number of nodes. Let

$$\textsc{Bipartite} = \big\{G \mid G \text{ is a bipartite graph}\big\}.$$

Show that $\textsc{Bipartite} \in$ NL. (Xi: What is the more natural complexity class to which $\textsc{Bipartite}$ belongs? as suggested by the fact mentioned, and what do we know about this class?)

3. Problem 9.19 on page 332 of TC (10 points): Show that if NEXPTIME $\neq$ EXPTIME then P $\neq$ NP. You may find the function pad, defined in Problem 9.18, to be helpful. Recall that

$$\text{EXPTIME} = \bigcup_{k>0} \text{TIME}\left(2^{n^k}\right) \quad \text{and} \quad \text{NEXPTIME} = \bigcup_{k>0} \text{NTIME}\left(2^{n^k}\right).$$

4. (10 points) Show that P $\neq$ SPACE(n). (Hint: Assume P = SPACE(n). Use the space hierarchy theorem to derive a contradiction. You may again find the function pad above useful here.)

5.* Problem 2.8.12 on page 55 of CC (7 points): Show that if a Turing machine (with a read-only input tape and any fixed number of working tapes, see the definition on page 35 of CC) uses space that is smaller than $c \log\log n$ for all $c > 0$, then it uses constant space. (Consider a "state" of a machine with input to also include the string contents of the work strings. Then the behavior of the machine on an input prefix can be characterized as a mapping from states to sets of states. Consider now the shortest input that requires space $S > 0$; all its prefixes must exhibit different behaviors – otherwise a shorter input requires space S. But the number of behaviors is doubly exponential in S.) (Xi: This problem is very closely related to Problem 4 of Set 1. The solutions to Set 1 will be posted on the course website, and you may want to have a look just to understand the hint above better.)

(3 points) Next, use the following example to show that the result above is tight. Let

$$L = \{b(1)\#b(2)\#\cdots\#b(k) : k \geq 1\},$$

where $b(i)$ denotes the binary representation of integer i and # is a special input symbol (so $L \subset \{0, 1, \#\}^*$). Describe (informally and concisely, no need to give all the details of your TM) a Turing machine that decides L with space $O(\log\log n)$ when the input string is of length n. (You do not have to solve the first part to work on the second part.)

6.* Problem 8.15 on page 303 of TC (5 points if you can show the following problem is in PSPACE; 10 points if you can show it is in P!): 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 the two players ever simultaneously reach positions that they previously occupied. Let

$$\textsc{Happy-Cat} = \Big\{(G, c, m, h) \mid G, c, m, h \text{ are respectively a graph, and positions of the Cat, Mouse, and Hole, such that Cat has a winning strategy, if Cat moves first.}\Big\}$$

Show that Happy-Cat is in PSPACE (and in P for more points).