CS 2110 Final Exam
Fall 2013

Directions

1. The test is closed book and closed notes.
2. Answer at most 6 part B questions.
3. Answer at most 3 part A questions.
4. One quarter credit will be given for unanswered questions. Answers that are way off the mark will get no credit.
5. For the part A questions, usually it is then a good idea to have a paragraph giving an overview of the proof strategy/technique that you will use, and what the key ideas are, before launching into the details.

PART B Questions

1. Define entropy and state Shannon's source coding theorem.
2. State Ladner's theorem.
3. Formally define pseudo-random generator.
4. State Landauer's principle. Give one situation where the principle is applicable, and one situation where the principle is not applicable.
5. Explain why the language of isomorphic graphs is in NP. Explain why it is unlikely that this language is NP-complete.
6. Which of these complexity class are known to have complete languages under that natural reductions: NP, co-NP, RP, PSPACE, P, NP $\cap$ co-NP, Σ_2^p? Explain what is different about the complexity classes that are known to have complete languages and those that are not known to have complete languages.
7. Give the Walsh-Hadamard code for 110. Show your work.
8. In the proof of Godel's incompleteness theorem, we showed that finding proofs for particular types number theoretic statements is not possible. Explain intuitively what these number theoretic statements are.
9. On which principle did the security of the quantum cryptographic protocol that we considered in class rest? State this principle. Explain in one sentence how this principle is related to security.
10. Draw a Venn diagram explaining the known inclusion relationships for the complexity classes: EXPONENTIALSPACE, P, EXPONENTIALTIME, LOGSPACE, PSPACE. State two general theorems from which all the inclusions follows from. State which inclusions are known to be proper, and state two general theorems from the properness of these inclusions follow from.
11. What is the standard circa 1970 proof technique to show that one complexity class properly contains another? Explain why it is believed that this technique won't be able to resolve the $P = NP$ question.
12. Define perfect secrecy of a private key cryptographic protocol.

PART A Questions

1. Consider a programming language mini-Java that only has one type of loop, and the number of iterations of the loop must be determined when the loop is first encountered. So a loop statement might look like "Repeat n times", and the variable n is evaluated when the statement is reached. You can assume that the program has a variable Size that is instantiated to the input size when the program starts running (otherwise, you couldn't even read the input). So all mini-Java programs must halt on all inputs. Show by diagonalization that there is a language accepted by a Java program that is not accepted by any mini-Java program. Use Diagonalization. Give a concrete example of a language that is not acceptable by any mini-Java program, but that is accepted by a Java program.

2. Explain the setup of the half-silvered mirror experiment. Explain how this is formally modeled within quantum mechanics. That is, how is the direction of the photon modeled, how are the mirrors modeled, and how is measurement modeled. Then show that calculations within this model give the result of the experiment in the real world.

3. Consider the following game played by two players A and B on a directed graph G with a designated start vertex s. On the first move, player A picks an edge (s, v) leaving s. Then s and v are designated as visited. On each even numbered move, player B picks an edge (v, w) from the current vertex v to a vertex w hasn't been visited before. If no such vertex w exists, then player B loses. Vertex w then is designated as visited and becomes the current vertex. On each odd numbered move, player A picks an edge (v, w) from the current vertex v to a vertex w hasn't been visited before. If no such vertex w exists, then player A loses. Vertex w then is designated as visited and becomes the current vertex. So the players A and B taking turns picking edges in a directed path P starting at s, with a player losing if he/she can't extend the path. Prove that determining for a particular G and s, whether the first player has a winning strategy is PSPACE-complete under polynomial time reductions. You must prove hardness by reduction from the problem of determining whether a quantified Boolean formula F is true.

4. Prove that determining the language of satisfiable Boolean formulas is NP-complete from first principles. That is, prove Cook's theorem.

5. Give an interactive protocol for proving the lower bound on the size of a set (within a factor of 2). You need only define pairwise independent hash function, but you need not explain how to find them efficiently. Prove that the protocol is correct with high probability.

6. State the PCP theorem, and then use the PCP theorem to prove that there exists an ϵ such that it is NP-hard to approximate MAXSAT within a factor of $1 + \epsilon$.