College of Computer & Information Science | Spring 2009
Northeastern University | Handout 14
CSG714: Theory of Computation | 13 March 2009

Take-Home Midterm Exam

Due at 3:25 PM, Friday, March 20

- **Honor code:** This exam is open-notes, open-library, and open-web. However, *no collaboration of any kind is allowed.* ("Collaboration" includes, for example, discussion or exchange of material related to the problems on the exam with anyone other than the instructor.) If you have any questions or clarifications, please ask me.
- **Policy on Cheating:** Students who violate the above rules on scholastic honesty are subject to academic and disciplinary penalties. Any student caught cheating will receive an **F** (failing grade) for the course and the case may also be forwarded to the Office of Student Conduct & Conflict Resolution for further disciplinary action.
- **Presentation of solutions:** While describing a proof or a solution, you may use any of the proofs and techniques covered in class or in the text by referencing the appropriate material, without elaboration. If you use any other sources, please include citations and write the solutions in your own words.
- **A note on grading:** Your grade on any problem will be determined on the basis of the correctness of your solution and the clarity of the description. Show your work, as partial credit will be given.
- **Best of luck!**

1. (15 points) Unary regular languages

Show that a language L over the unary alphabet $\{0\}$ is regular if and only if there exist integers r, $r_1, r_2, \ldots, r_j, r_{j+1}, \ldots, r_k$ (for some finite j and k) such that

$$L = \left(\bigcup_{1 \le i \le j} 0^{r_i} \right) \bigcup \left(\bigcup_{j < i \le k} (0^r)^* \cdot 0^{r_i} \right).$$

2. (15 points) Input-uniform Turing machines

We say that a deterministic single-tape Turing machine is *input-uniform* if the position of its head at the ith step of its computation depends only on i and $|x|$, not x itself.

Prove or disprove: The set of languages recognized by input-uniform Turing machines is exactly the set of Turing-recognizable languages.

3. ($4 \times 10 = 40$ points) Classifying languages

Each of the following three parts gives a definition, description, or some properties of a language. In each case, tell whether the language is:

A regular

B context-free but not regular

C Turing-decidable but not context-free

D Turing-recognizable but not Turing-decidable

E not Turing-recognizable

F there is insufficient information to tell.

Justify your answers with proofs, constructions, algorithms, or examples as needed.

If, for example, you decide that a language is context-free but not regular, you must give a context-free grammar that generates the language or a pushdown automaton that accepts the language and also prove that the language is not regular.

(a) The set of all Turing machine descriptions $\langle M \rangle$, where M is a nondeterministic Turing machine and there exists a string x such that M halts on x on all computation paths.

(b) $\{\langle M \rangle$: the language recognized by M is finite$\}$

(c) $\{0^{f_1}1^{f_2}0^{f_3}1^{f_4}\ldots 0^{f_k} : k, f_1, \ldots, f_k$ are positive integers and f_i is the ith Fibonacci number$\}$

(d) The complement of the language of part (c).

4. (3 + 12 = 15 points) Pirates Undercover

My daughter has a fun puzzle called *Pirates Undercover*[1]. The puzzle consists of a 2×2 game board, 4 puzzle pieces, and several puzzle challenges. Each of the four cells of the game board has some pictures on it; each picture is one of five types – a pirate ship, a white sailboat, a rowboat, an island, and a cave. The pictures and their positions on the cell differ from cell to cell. Each of the four puzzle pieces can be placed on a cell in multiple (but a finite number of) orientations, and each orientation hides a subset of the pictures. Each puzzle challenge consists of a count of the number of pictures of each kind. The goal is to place one puzzle piece on each cell so that the collection of all visible pictures match the prescribed count.

Let us generalize the above puzzle. Suppose, we are given a game board with n cells. Each cell has some pictures, each picture being of one of k types. We are also given n puzzle pieces; each puzzle piece can be placed on a cell in a finite number of orientations (say ℓ), each orientation hiding some pictures and showing some. Given a count for each of the k picture types, we are asked whether there exists a placement of the puzzle pieces on the game board (one piece on each cell) such the for each picture type, the number of visible pictures equals the count for the type.

(a) Formulate the above problem as a language and show that it is in NP.

(b) Show that the language is NP-hard.

[1] http://www.toys4minds.com/pirates-undercover-game-educational-insights.html

5. (5 + 10 = 15 points) Business Games

Consider the following game between two companies in a tough zoning environment. The game is specified by an undirected graph $G = (V, E)$, a function $r : V \to Z$, and an integer R. The graph G is the graph of divisions in a city, each vertex being a division and each edge representing two neighboring divisions. Each vertex v in the graph has an integer $r(v)$ associated with it, indicating the revenue that a firm will obtain by setting up a store at v. Zoning laws require (a) at most one of the companies can open a store at any division v, and (b) the set of all store locations (together, by both the firms) forms an independent set (i.e., if A or B opens a store at a division v, then neither of the companies can open a store at any of the divisions adjacent to v).

The game proceeds in a sequence of steps. Firm A goes first, opens a store at a particular division. Then firm B opens a store, satisfying the constraints mentioned above. Then A goes next, and so on. The goal of B is to open stores so as to attain a total revenue of at least a given bound R.

Let L be the set of all games $\langle G, r, R \rangle$ in which B achieves a revenue of R.

(a) Show that L is in PSPACE.

(b) Show that L is PSPACE-hard.

(*Hint:* Reduce from the Formula Game discussed in class and also defined in Sipser's text (Chapter 8). You may assume that the SAT formula in the formula game is a 3-SAT formula. Associate vertices of the graph with the variables and clauses in the formula. Make sure that you assign the revenues such that the moves the players make in the competition game correspond to assigning truth values to the variables in the formula in the appropriate order.)