CS 517
Due Wednesday, 1 June

Homework #8

1. An **equivalence relation** (usually represented as $\equiv$ or $=$) on a set S must have the 3 properties of *reflexitivity*, *transitivity*, and *symmetry*. A **partial order** (usually represented as $\leq$) must have the properties of *reflexitivity*, *transitivity*, and according to Knuth also *anti-symmetry*, defined by

$$x \leq y \qquad \text{and} \qquad y \leq x \qquad \text{imply that} \quad x = y.$$

Consider the usual ordering on the positive rationals:

$$\frac{a}{b} \leq \frac{p}{q} \qquad \text{iff} \qquad a \cdot q \leq p \cdot b.$$

Does

$$\frac{a}{b} \leq \frac{p}{q} \qquad \text{and} \qquad \frac{p}{q} \leq \frac{a}{b} \qquad \text{imply} \qquad \frac{a}{b} = \frac{p}{q}?$$

Thge standard method to enforce *anti-symmetry* uses **equivalence classes**. Define $x \equiv y$ (corresponding to $\leq$) iff $x \leq y$ and $y \leq x$. Now apply $\leq$ to the equivalence classes defined by this $\equiv$. Let $[x]$ mean the equivalence class which contains x.

(a) Show that $\leq$ is **well-defined**, that is,

$$x \leq y \quad \text{iff} \quad [x] \leq [y].$$

(There is actually something to show here. Is the claim **TRUE** in the extreme case when the set S is null ?)

(b) Show that $\leq$ on equivalence classes is reflexive and transitive.

(c) Show that $\leq$ on equivalence classes is anti-symmetric even if $\leq$ on elements is not.

(d) If the positive rational numbers obey a partial order (in the above sense), is it reasonable to say that a rational number is an ordered pair of positive integers?

(e) An ordering is sometimes called a *total order* if it is a partial order and obeys the TRICHOTOMY law:
For every pair of elements x and y exactly one of the following is true

$$x \le y \quad \text{or} \quad y \le x \quad \text{or} \quad x = y.$$

Fix this definition so that it makes sense with our definition of $\le$ as a partial order

(f) Show that the rationals obey your definition of total order.

(g) Give an example of a partial order which is **NOT** a total order. (What about $\mathcal{NP}$ using $\le_{Karp}$?)

2. Use examples to distinguish between each of the folowing orderings:
“ is a *subproblem* of ”
“ is a *subset* of ”
“ is a no harder than ”.
Do any of these relations imply any of the others?
E.G., if $A \ \le_1 \ B$ for one of these orderings $\le_1$, does this imply that $A \ \le_2 \ B$ for one of the other orderings $\le_2$?

3. Discuss: Do either of the following relationships hold?

$$A \le B \implies \overline{A} \le \overline{B}$$

$$A \le \overline{B} \implies \overline{A} \le B\ .$$

4. Define *log-space* reduction by

$$A \ \le_{\log} \ B$$

iff there is a log-space computable function f, so that for all x,

$$x \in A \qquad \textbf{iff} \qquad f(x) \in B.$$

(a) Show that $\le_{\log}$ is transitive.

(b) Give an example of two sets A and B, so that

$$A \ \le_{\log} \ B.$$

5. Define *linear-space* reduction by

$$A \leq_{LIN} B$$

iff there is a linear-space computable function f, so that for all x,

$$x \in A \qquad \textbf{iff} \qquad f(x) \in B.$$

   (a) Is $\leq_{LIN}$ transitive?

   (b) Give an example of two sets A and B, so that $A \leq_{LIN} B$.

   (c) Give a third set C, so that $B \leq_{LIN} C$. Is $A \leq_{LIN} C$?

6.

$$SHALT = \{ (M, x, (1)^t) \mid M \text{ halts and uses at most } t \text{ tape squares when given } x \}$$

$(1)^t$ means a string consisting of t 1's.
Show that $SHALT$ is $\mathcal{PSPACE}$-complete.

7. Assume that the behavior of a probabilistic algorithm can be modeled using a natural number n, so that the algorithm terminates when n reaches 0. Further, assume that with probability β the algorithm transitions from n to $n+1$, and with probability $1-\beta$ the algorithm transitions from n to $n-1$.
Let p_n be the probability that the algorithm terminates when started at n. The following difference equation gives a relation among the p_n's

$$p_n = \beta\, p_{n+1} + (1-\beta)\, p_{n-1}.$$

   (a) Show that if $\beta > 1/2$ then $p_n = \left(\frac{1-\beta}{\beta}\right)^n$ is a solution of the difference equation.

   (b) Does the above assure that the algorithm always terminates?

   (c) If $\beta < 1/2$ what happens to the above solution, and how does the algorithm behave?

8. Show that $\{ a^n \, b^{n^2} \}$.

   (a) What is the simplest type of grammar which can generate this language?

   (b) Show that no simpler type of grammar can generate this language.

9. **CSL-Member**
   INPUT: G a context-sensitive grammar, and x a string.
   QUESTION: Can G generate x?

   It is known that **CSL-Member** is a $\mathcal{N}$-LinearSpace complete set.

   (a) What is deterministic complexity class is **CSL-Member** in?

   (b) Use the above to explain why programming languages are described using context-free grammars rather than context-sensitive grammars.

10. Let M^{R} be the deterministic finite state machine which recognizes the *reverse* of the language recognized by the deterministic finite state machine M.
    Let M be the Towers of Hanoi machine from the "Moore Reducibility" paper.

    (a) Construct M^{R}.

    (b) Compare the number of states of M^{R} and M.

    (c) Find the level in the Moore hierarchy for the language recognized by M^{R}.