Algorithmic Game Theory	**Problem Set 3**
CS 6840 Spring 2010	**Due Monday, April 5th**

We will maintain a FAQ for the problem set on the course Web page. You may use any fact we proved in class without proving the proof or reference, and may read the relevant chapters of the book. However, you may not use other published papers, or the Web to find your answer.

A full solution for each problem includes proving that your answer is correct. Please start by explaining what is the high-level idea of the solution (the main insights/nontrivial things necessaries to solve them problem). If you think its useful you may add also pseudocode for details. Do not submit a code only. It can make the solutions more readable if you introduce convenient notation and use it.

If you cannot solve a problem, write down how far you got, and why are you stuck. You may solve the problems with a partner, and may use a different partner for each problem set (though not a different partner for each individual problem). Please hand in a shared problem set with both your names on the solutions. We are happy to help finding partners, let us (Renato or Hu) know if you don't have a partner, and would like one.

Solutions can be submitted on CMS, or handed in at Renato or Hu's office. Please type your solution, to make it easier to read.

(1) We know that in a game with a finite set of players, where each player has a finite set of pure strategies, the game has a Nash equilibrium. The standard proof is based on Brouwer's fixed point theorem. Unfortunately, the proof of the fixed point theorem is not algorithm. We'll talk later in the course about the harness of this problem. In this problem, we explore if one can at least do this in finite time (maybe exponential). Consider the special case with two players. To be more formal, assume that there are 2 players, and player i chooses between n_i pure strategies. Assume that the game is given by the matrices A and B, listing the payoffs for the two players respectively for each $n_1 \times n_2$ possible plays. This is the traditional payoff matrix that is traditionally called payoff matrix.

(a) Give a polynomial time algorithm to check if there is a Nash equilibrium strategy for the game in which each player mixes between at most two strategies.

(b) Give a finite algorithm for finding a Nash equilibrium for general games with two players. Your algorithm may run in exponential time.

(2) Consider a two player game with two reward matrices A and B as also used in the previous problem, and assume that both players have n possible strategies (so A and B are n by n matrices. Assume that the matrix A and B has random entries, say all entries in the range $[0, 1]$ filed out uniformly independently at random. Show that the provability that this random game has a pure (deterministic) Nash equilibrium is at least roughly $1 - 1/e$ if n is large. You may use the fact that for large n we have that $(1 - 1/n)^n \approx 1/e$.

Warning. You may want to compute the probability that a pair of strategies (i, j) forms a Nash. Unfortunately, these events are **not** independent!

(3) Problem 20.5 in the book considers the load balancing game, where each job j has a "processing time" or weight w_j and each machine i has a speed s_i, where the load job j presents if assigned to machine i is w_j/s_i. A Nash equilibrium is fully mixed, if for each job j and each machine i the probability π_{ij} of assigning job j to machine i is positive.

(a) Solve problem 20.5 in the book: show that such a load balancing game can have at most one mixed equilibrium

(b) Give an example of a load balancing game that has no fully mixed equilibria.

(4) Are the set of coarse correlated equilibria is larger than the set of correlated equilibria for two player games with 2 strategies each? How about games with 2 players and 3 strategies each? Can the average payoff be also different?

(5) An action s_i of a player i is ϵ-dominated by action s'_i for all strategy profiles s_{-i} of the other players $u_i(s_i, s_i) \leq u_i(s'_i, s_i) - \epsilon$. Let s_i be an ϵ-dominated action of a player i.

(a) Show that if a player i uses the weighted majority algorithm discussed in class to choose his/her strategies, that the probability $\pi(s_i)$ that he/she is playing strategy s_i goes to zero over time.

(b) Give an example of a game with a coarse correlated equilibrium, and an ϵ-dominated action of a player i, where player i is playing action s_i with positive probability.

(c) Can this also happen in correlated equiliria? (i.e., can there be a correlated equilibrium when player i plays his/her ϵ-dominated action with positive probability?

(6) Hotelling games is a general class of games when k providers for a set of customers. Here we use the following simple case: G is a graph on n vertices. There are k providers, and each provider selects one of the nodes of the game, you can think of the location as a souvenir stand. Once the sellers selected their locations. Each node v in the graph has $n_v > 0$ customers (tourists), and each costumer selects the closest seller. In case of ties divide the n_v costumers uniformly among the closest sellers (OK if the fractions are not integers). The goal of the sellers is to attract as many buyers as possible. Let N_i be the total number of customers who selected seller i. In this game the traditional social welfare is not a good measure, as I assumed all customers choose a seller, and hence $\sum_i N_i = \sum_v n_v = N$. Instead we will look at a fairness measure, $\min_i N_i$.

(a) Show that the price of anarchy for pure Nash equilibria in this game is bounded by 2. By which we mean that if *Opt* denotes the value of the most fair allocation $\max \min_i N_i$, where the maximum is taking over the possible locations of the k sellers, then for any pure strategy Nash equilibria $\min_i N_i > Opt/2$.

(b) In this game the utility of a player i is between $[0, N]$. The weighted majority algorithm assumed utilities are in the range $[0, 1]$. Show how to adopt the weighted majority algorithm for this game.

(c) Show that if we play this game repeatedly, and a player i player used a no-regret algorithm, than this payer is guaranteed to get $N_i \geq Opt/2$ customers, independent of the strategies used by other players. More precisely, assume that the player has small total regret over T steps at most ϵTN, then he/she is guaranteed an average value at least $Opt/2 - f(\epsilon)$, where $f(\epsilon)$ goes to zero as ϵ goes to zero.