14.12 Game Theory
Muhamet Yildiz
Fall 2005

Homework 3

Due on 10/31/2005

1. Two players first simultaneously declare Yes or No. If any of the player declares No, then each player gets 80. If both players declare Yes, then they play the following game:

	Invest	Consume
Invest	100, 100	0, 60
Consume	60, 0	60, 60

   Find all the subgame-perfect equilibria of the entire game.

2. Consider the bargaining game discussed in the class, where two players try to divide a dollar, which they cannot consume until they divide, and the payoff from getting share x at t is $\delta^{t-1}x$. At each date a player proposes a division $(x, 1-x)$, and the other player accepts or rejects. If the offer is accepted, then it is implemented, ending the game; otherwise, we proceed to the next date. The order of proposers is Player 1, Player 1, Player 2, Player 1, Player 1, Player 2, Player 1, ..., i.e., Player 1 makes two offers for each offer of Player 2.

   (a) Assume that the game ends at date $3n$, after which players get 0. Use backwards induction to compute an equilibrium.

   (b) Assume that there is no deadline so that the game ends only when a player accepts an offer. Compute a subgame-perfect equilibrium.

   (c) Now instead assume that at each date we determine the proposer by tossing a coin, so that the probability that a player i will make an offer at date t is $1/2$, and this probability does not depend on the past histories. Repeat parts (a) and (b).

3. There are n firms.

   - First, simultaneously each firm decides whether to enter a market, by incurring a cost C. (If a firm does not enter, its payoff is 0.)
   - Then, knowing which firms entered, each firm i in the market simultaneously produces q_i at zero marginal cost, and they sell at price $P = \max\{1-Q, 0\}$, where Q is the sum of the q_is produced by the firms in the market. The payoff of a firm i in the market is $q_i P - C$.

   Find all the subgame-perfect equilibria in pure strategies.

4. Goliath Software is a large software company that sells an internet browser called X. Each year a new startup comes along. The startup can either produce another browser that is comparable to X or a search engine. Seeing what the startup produced, Goliath

Software either updates X or not. At the end of that year, independent of the outcome, the start up disappears, leaving its place to the next start up. The annual profits for each contingency are as in the following table, where the first entry is the profit of Goliath Software:

	Browser	Search Engine
Update	1,0	3,1
No Update	2,2	4,1

(The above is just a table, not a game.) There are 1000 years. The payoff of Goliath software is the sum of its annual profits. The payoff of each startup is its own annual profit. The entire history is observable. Find all the subgame-perfect equilibria.