Financial Software Engineering

Take Home Exam 2019

Instructions

- This exam follows the honor code. You are not allowed to work together with other students and all answers must be in your own words. If answers are not in your own words, I reserve the right to deduct some or even all marks.
- The submission deadline is 18:00 on 21 August 2019. Late submissions will have marks deducted.
- For Section 2, you may use resources such as packages, CDN's and templates.
- Answer all questions including all sub-questions. There are 100 points in total and each question has a clearly indicated number of points.
- Upon completion of the take-home, please zip the folder and send it to co-pierre.georg@uct.ac.za

---

Section 1 (Total: 30 Marks)

Question 1.1: Consider the following two finite versions of the Cournot duopoly model. First, suppose each firm must choose either half the monopoly quantity, q_m/2 = (a-c)/4, or the Cournot equilibrium quantity, q_c = (a-c)/3. No other quantities are feasible. Show that this two-action game is equivalent to the Prisoner's Dilemma: each firm has a strictly dominated strategy, and both are worse off in equilibrium than they would be if they cooperated. Second, suppose each firm can choose either q_m/2 or q_c, or a third quantity, q'. Find a value for q' such that the game is equivalent to the Cournot model discussed in class and in Gibbons, Section 1.2.A, in the sense that (q_c, q_c) is a unique Nash equilibrium and both firms are worse off in equilibrium than they could be if they cooperated, but neither firm has a strictly dominated strategy. **[6 marks]**

Question 1.2: Three oligopolists operate in a market with inverse demand given by P(Q) = a - Q, where Q = q_1 + q_2 + q_3 and q_i is the quantity produced by firm i. Each firm has a constant marginal cost of production, c, and no fixed cost. The firms choose their quantities as follows: (1) firm 1 chooses q_1 >= 0; (2) firms 2 and 3 observe q_1, and then simultaneously choose q_2 and q_3, respectively. What is the subgame perfect outcome. [**4 marks]**

Question 1.3: What is a strategy in a repeated game? What is a subgame in a repeated game? What is a subgame-perfect Nash equilibrium? **[3 marks]**

Question 1.4: Give the extensive-form and normal-form representations of the bank-runs game discussed in class and in Gibbons Section 2.2.B. What are the pure-strategy subgame-perfect Nash equilibria? **[3 marks]**

Question 1.5: Consider the following asymmetric-information model of Betrand duopoly with differentiated products. Demand from firm i is $q_i(p_i,p_j) = a - p_i - b_i*p_j$ (i and j are hard to distinguish in subscripts, so I colored the p_j blue for your convenience). Costs are zero for both firms. The sensitivity of firm i's demand to firm j's price is either high or low. That is, b_i is either b_h or b_L, where $b_H > b_L > 0$. For each firm, $b_i = b_H$ with probability h and $b_i = b_L$ with probability 1-h, independent of the realization of b_j. Each firm knows its own b_i but not its competitor's. All of this is common knowledge. What are the action spaces, type spaces, beliefs, and utility functions in this game? What are the strategy spaces? What conditions define a symmetric pure-strategy Bayesian Nash equilibrium of this game? Solve for such an equilibrium. **[6 marks]**

Question 1.6: Consider a first-price, sealed-bid auction in which the bidders' valuations are independently and uniformly distributed on [0,1]. Show that if there are n bidders, then the strategy of bidding (n-1)/n times one's valuation is a symmetric Bayesian Nash equilibrium of this auction. **[4 marks]**

Question 1.7: Consider a first-price, sealed-bid auction in which the bidders' valuations v_i are independently and identically distributed according to a strictly positive density $f(v_i)$ on [0,1]. Compute a symmetric Bayesian Nash equilibrium for the two-bidder case. **[4 marks]**

Section 2 (Total: 70 Marks):

As you know, Co is obsessed with sneakers. He however struggles to keep track of what he purchases, when these purchases are made and how much he spends. Ideally, there is some application to do this.

You are required to build such an application, including the HTML, CSS and JavaScript. You are provided with a (i) JSON file with all of Co's sneaker purchases so far and a (ii) JS file with the sneaker information already loaded as a variable. You can choose to make use of either one. In other words, (i) you can read in the JSON file into your project, or (ii) use the JS file with the sneaker data already stored as a variable. The file contains the following variables

- Brand: Brand of the sneaker
- Style: The style of the sneaker
- Color: Color of the sneaker
- Date: Date the sneaker was purchased
- Price: Price paid in South African Rand

You are required to develop a single web page which allows Co to view with his sneaker purchase history. The page should have two main features.

1. The user should be able to view their existing sneaker collection

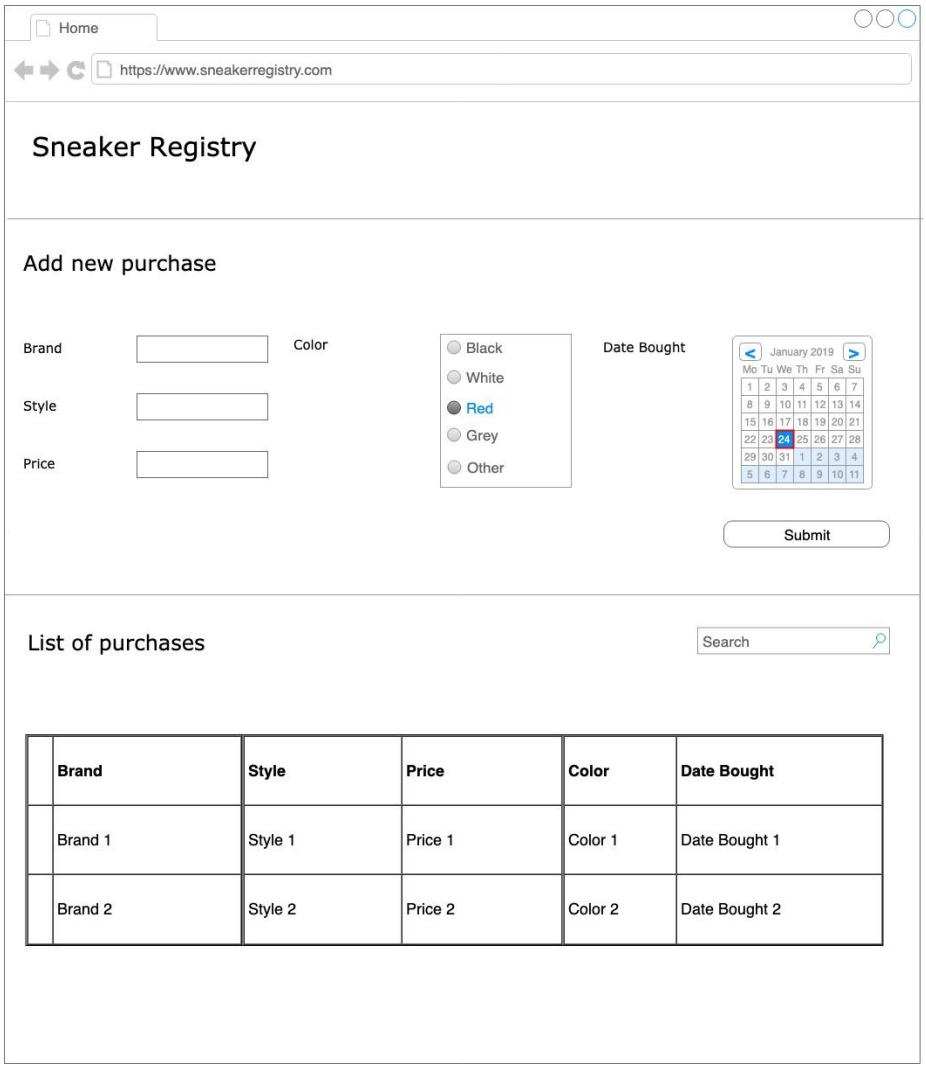
- Present the user's sneaker information in tabular form or any other representation you deem appropriate
- The table should be searchable and allow the user to sort on columns

2. The user should be able to add purchases to their sneaker collection

- Each time a user purchases a new pair of sneakers, they should have some way to record these transactions.
- You should provide the functionality to accept the required information
- Importantly, make sure that the user provides the relevant information in the correct format
    - For example, the user cannot include an alphabetic character in the price field
    - Print appropriate messages to the user when the information provided does not meet the required format
    - Also, print an appropriate message if the user fails to provide certain information
    - Only allow the user to submit information once it adheres to the required format

- Once the user submits the required information, append it to the table showing the user's sneaker collection
- **Note**: you do not need to implement the contents of the JSON file as a database and all variables can live locally in the JS browser.

A mock-up of a possible layout is provided below



Mark allocation

- 50 marks are allocated for the implementation of the required functionality
- 20 marks are awarded for the aesthetics of the webpage and the general structure of your HTML, CSS and JS code

Bonus question (Total: 10 Marks)

Implement the JSON file in Section 2 as a database. You should read the JSON file in using Javascript. Every time a sneaker purchase is made it should be appended to the JSON file. **Tip**: this will require you to use node.js and host a local server