Problem Set #1

MACS 40000, Dr. Evans
Due Wednesday, Sep. 27 at 1:30pm

1. **Git and GitHub.com (3 points).**

   (a) Make sure Git is installed on your computer.

   (b) Sign up for a GitHub.com account. I recommend that you choose a username that you would be happy for people to know you by when you become famous later in life. Be professional.

   (c) Select "Watch" in the GitHub repository for this class at the following URL: https://github.com/rickecon/OGcourse_F17.

   (d) Make a "fork" of this repository in your own GitHub account, and "clone" your fork (not my repository) to your local machine.

2. **2-period-lived OG economy (7 points).** Your algebra, analysis, and answers to the following questions should be typeset using LaTeX. Consider a two-period-lived agent overlapping generations economy in which a unit measure of agents are born every period, they have identical preferences, and they each receive the same endowment of consumption good every period. The economy starts in period $t = 1$ with an old agent and a newly born young agent.

   Agents born in period $t \geq 1$ choose consumption $c_{s,t}$ in each age of life $s = \{1, 2\}$ and each period t to maximize lifetime utility,

   $$\max_{c_{1,t}, c_{2,t+1}} (1 - \beta) \ln\left(c_{1,t}\right) + \beta \ln\left(c_{2,t+1}\right) \quad \forall t \tag{1}$$

   where $c_{1,t}$ is consumption by the age $s = 1$ individual in period t, $c_{2,t+1}$ is consumption by that same individual in the last period of life $s = 2$, and $\beta \in (0, 1)$ determines how heavily the household weights consumptions when old relative to consumption when young.

   Each individual is endowed with a measure of consumption good $e_1, e_2 > 0$ for all t that is perishable (nonstorable). However, this consumption good can be traded between old and young in a given period for price p_t. Therefore, an agent born in period t has the following lifetime budget constraint.

   $$p_t c_{1,t} + p_{t+1} c_{2,t+1} \leq p_t e_1 + p_{t+1} e_2 \tag{2}$$

   The initial old are fundamentally different from every other generation in that they simply choose consumption in their last period of life to maximize a one-period utility function.

   $$\max_{c_{2,1}} \beta \ln\left(c_{2,1}\right) \quad \text{s.t.} \quad p_1 c_{2,1} \leq p_1 e_2 \tag{3}$$

A competitive equilibrium is allocations $c_{1,t}$ and $c_{2,t}$ and prices p_t for all $t \geq 1$ characterized by all individuals in every period choosing consumption to maximize lifetime utility subject to their lifetime budget constraint and markets must clear. The market clearing condition in every period is the following.

$$c_{1,t} + c_{2,t} = e_1 + e_2$$

For agents born in periods $t \geq 1$, they choose consumption $c_{1,t}$ and $c_{2,t}$ to maximize (1) subject to (2). For the initial old in period 1, they choose old-age consumption $c_{2,1}$ to maximize old-age utility subject to their one-period budget constraint in (3).

(a) Solve for the household's (born in period $t \geq 1$ optimal consumption when young $c_{1,t}$ and when old $c_{2,t+1}$ given prices $p_t > 0$ and $p_{t+1} > 0$, preferences β, and endowments $e_1 > 0$ and $e_2 > 0$. [Note: I want the partial competitive equilibrium solution to the 2-period household problem.]

(b) Solve for the initial old person's optimal consumption when old $c_{2,1}$ given price $p_1 > 0$ and endowment $e_1 > 0$ by solving the optimization problem (3). [Note: I want the partial competitive equilibrium solution to the 1-period household problem.]

(c) Solve for the competitive equilibrium of consumptions $\{c_{1,t}, c_{2,t}\}_{t=1}^{\infty}$ and prices $\{p_t\}_{t+1}^{\infty}$ with the normalization $p_1 = 1$ given the household optimization and market clearing. [Hint: You will have to use the household's maximization problem in (1) instead of your answer to part (a).] Does this equilbrium equal your answer to part (a)?

APPENDIX

A.1 Git Instructions

Once you have Git is installed on your machine and you have your GitHub.com account, you can clone your own fork of the class GitHub repository by completing the following steps.

1. Go to the class GitHub repository at https://github.com/rickecon/OGcourse_F17.
2. "Fork" this repository by clicking on the "Fork" button in the upper-right of the screen.
    - GitHub will ask you "Where do you want to fork this repository?" Select your account.
    - Forking the repository makes a copy of this repository in your account.
3. Clone your fork of the repository by doing the following steps:
    - Go to the website of your fork of the repository by navigating to the URL: https://github.com/YourUserName/OGcourse_F17
    - Click on the green "Clone or download" button and copy the URL.
    - In the terminal on your computer, navigate to the directory where you want to place this repository on your local machine and type: `git clone [the copied URL]`
    - Navigate to that new directory in your terminal by typing: `cd OGcourse_F17`.
4. For ease of updating your fork we want to give the main course repository the designation of "upstream".
    - Go to the class repository at https://github.com/rickecon/OGcourse_F17, click the green "Clone or download" button, and copy the URL to your clipboard.
    - In your terminal, navigate to the directory of your local clone of your fork of the class repository.
    - Type: `git remote add upstream [the copied URL]`.
    - You can now update your repository by typing: `git fetch upstream`, followed by: `git merge upstream/master`, and lastly: `git push origin`.
5. Commit your work and changes to your GitHub fork often by doing the following steps.
    - Type: `git add -A` or type: `git add [specific files]`.
    - Type: `git commit -m "[message describing commit]"`.

- Type: `git push origin master`.

6. When you are ready to submit your assignment, turn it in by submitting a pull request to the main class repository.

    - Make sure you have created a subfolder "OGcourse/ProblemSets/[LastName_FirstInit]" that has your Python code and LaTeXwriteups of the problem sets with all your changes committed and pushed.
    - Go to the web page of your fork of the repository, https://github.com/YourUserName/OGcourse_F17 and click on "New pull request".
    - Submit the pull request with any commentary or questions you might have.
    - If you submit your code early, this becomes a nice environment for me to give online help with your code.

A.2 LaTeXInstructions

1. Download a LaTeXdistribution to your machine.

    - I suggest MacTeX for Mac OS X, MiKTeX for Windows, and TeX Live for Linux.

2. Use a text editor to compile LaTeX code to PDF

3. I provide you with a template for writing LaTeX documents: OGcourse_F17/ProblemSets/PStemplate.tex.