

```
%% Macro III: Problem Set 3
% Deadline: Friday, 17/09/2018
%
% Aluno: Bruno Tebaldi de Queiroz Barbosa (C174887)
%
% Professor: Tiago Cavalcanti
%
% Source code disponível em: <https://github.com/btebaldi/Macro3/tree/master/PSet\_02
% https://github.com/btebaldi/Macro3/tree/master/PSet\_03>
%
%
%% Questao 2
%% Item F
% A arrecadação do governo pode ser escrita como:
%
%  $G = \tau_y K_t^{\alpha} N_t^{1-\alpha} + \tau_n w_t N_t$ 
%
% Em termos per capita temos:
%
%  $g = \tau_y k_t^{\alpha} + \tau_n w_t$ 
%
% Se a arrecadação é constante, pelo teorema da função implícita temos:
%
%  $\frac{d\tau_y}{d\tau_n} = \frac{-w}{k^{\alpha}}$ 
%
% substituindo a equacoes de w temos:
%
%  $\frac{d\tau_y}{d\tau_n} = -\frac{(1-\tau_y)(1-\alpha)}{(1+\tau_n)}$ 
%
%  $\frac{d\tau_y}{d\tau_n} = -0.48$ 
%
% Utilizando o conceito de diferencial temos:
%
%  $\Delta\tau_y = -0.48 \Delta\tau_n = (-0.48)*(-0.05) = 0.024$ 
%
% Logo é esperado que  $\tau_y$  seja próximo de 0.024.

% Lipeza de variaveis
clearvars
clc

% Cria um processo de markov com as características especificadas.
sigma = ((1-0.98^2)*0.621)^0.5;
mkv = MarkovProcess(0.98, sigma ,2,7,0);
mkv.AR.sigma2_y
[chain,state] = MarkovSimulation(mkv.TransitionMatrix, 1000, mkv.StateVector, 3);

eco_param.alpha = 0.4;
eco_param.beta = 0.96;
eco_param.delta = 0.08;
```

```
eco_param.gamma = 0.75;

eco_param.sigma_c = 2;
eco_param.sigma_l = 2;

eco_param.tau_n = 0.25;
eco_param.tau_y = 0;

% Determina os parametros r e w da economia
eco_param.r_UpperBound = 1/eco_param.beta -1;
eco_param.r_LowerBond = -eco_param.delta;
eco_param.r = (eco_param.r_UpperBound + eco_param.r_LowerBond)/2;
eco_param.r_tilda = eco_param.r + eco_param.delta;

% Determina os Grids
WageShocks.Values = exp(mkv.StateVector);
WageShocks.Grid.Min = min(WageShocks.Values);
WageShocks.Grid.Max = max(WageShocks.Values);
WageShocks.Grid.N = mkv.QtdStates;
WageShocks.PI = mkv.TransitionMatrix;

% Determina as caracteristicas do grid de trabalho
Labor.Grid.N = 20;
Labor.Grid.Min = 0.01;
Labor.Grid.Max = 1;
Labor.Values = linspace(Labor.Grid.Min, Labor.Grid.Max, Labor.Grid.N);

G0 = CalculaRendaGov(eco_param.tau_y ,Labor, WageShocks, eco_param, 1);

eco_param.tau_n = 0.20
f = @(tau_y) ( G0 - CalculaRendaGov(tau_y, Labor, WageShocks, eco_param, 0));

sol = fsolve(f,0.024);

CalculaRendaGov(sol, Labor, WageShocks, eco_param, 1);
```