

```
function out = FirstOrderCondition(z,param_sys,param_gov,param_conditions)
```

```
% leitura dos parametros do sistema
```

```
beta = param_sys(1);  
delta= param_sys(2);  
gamma = param_sys(3);  
alpha = param_sys(4);  
theta = param_sys(5);  
eta = param_sys(6);
```

```
% leitura dos parametros do governo
```

```
tau_c= param_gov(1);  
tau_h = param_gov(2);  
tau_k = param_gov(3);
```

```
% leitura do total de periodos do path
```

```
T=param_conditions(1);
```

```
% leitura do ponto inicial
```

```
k_0=param_conditions(2);  
% h_0=param_conditions(4);
```

```
% leitura dos parametros de steady state
```

```
k_ss=param_conditions(3);  
h_ss=param_conditions(5);
```

```
% Inicializacao das variaveis endogenas
```

```
% Sao T periodos +1 (condicao final)  
k=nan(T+1);  
h=nan(T+1);
```

```
% Leitura das variaveis endogeas (definidas pelo sistema)
```

```
% Dos T periodos
```

```
for i=1:T  
    k(i)=z(i,1);  
    h(i)=z(i,2);  
end
```

```
% No primeiro periodo o sistema esta com k=k_0 (condicao dada)
```

```
k(1)=k_0;
```

```
% No ultimo periodo o sistema deve estar em equilibrio.
```

```
% Logo as variaveis endogenas devem ser iguais a do steady state
```

```
k(T+1)=k_ss;  
h(T+1)=h_ss;
```

```
% Definicao das condicoes de primeira ordem
```

```
% Inicializacao da FOC's
```

```
% Temos T+1 FOC's pois o ultimo periodo nao tem FOC pois já esta em SS
```

```
f1 = nan(1,T);
```

```
f2 = nan(1,T);

for t=1:T
    %c(t)
    c_t = (1-alpha)* k(t)^alpha * h(t)^(-alpha) * ((1-tau_h)/(1+tau_c))^((1-h(t))/theta);

    %c(t+1)/c(t)
    c1_c = ((1-h(t+1))/(1-h(t)))*(k(t+1)/k(t))^alpha*(h(t)/h(t+1))^alpha;

    % FOC (1)
    f1(t)=((1-delta)*k(t)-(1+tau_c)*c_t) + k(t)^alpha*h(t)^(1-alpha)*(1 -tau_h*(1-alpha) ✓
    -tau_k*alpha)-(1+eta)*(1+gamma)*k(t+1);

    % FOC (2)
    f2(t)=-c1_c*(1+gamma) + beta*(1-delta + alpha*(h(t+1)^(1-alpha))*(k(t+1)^(alpha-1))* ✓
    (1 - tau_k));
end

% a saida da funcao sera um vetor de FOC's
out=[f1' f2'];

end % end of FirstOrderCondition
```