```
clear all
clc

% Parameters
syms J M R g To real
% Variables
syms x theta  real
syms dx dtheta real

% Define symbolic variable q for the generalized coordinates
% x and theta
q  = [x; theta];
% Define symbolic variable dq for the derivatives
% of the generalized coordinates
dq = [dx; dtheta];
% Write the expressions for the position of
% the center of the ball:
p = [x*cos(theta) - R*sin(theta); 0; x*sin(theta) + R*cos(theta)];

% Kinetic energy
T = 0.5*J*dtheta^2; % kinetic energy of beam

dp = jacobian(p,q);
T  = T + 0.5*M*dx^2; %((dx+dtheta*R)^2 + (dtheta*x)^2) ; % add linear
 kinetic energy of ball

% I     = 0.4*M*R^2; % inertia in rotation of ball
% omega = dtheta + dx/R; % angular velocity of ball

T  = T + 0.5*(2/5*M*R+M*p'*p)*dtheta^2 + 0.5*(2/5*M*R)*(dx/R)^2; % add
 rotational kinetic energy of ball

T = simplify(T);

% Potential energy
V = M*g*(R*cos(theta)-x*sin(theta));

% Generalized forces
Q = [0; To];

% Lagrangian
Lag = T - V;

Lag_q = simplify(jacobian(Lag,q)).';
Lag_qdq = simplify(jacobian(Lag_q.',dq));
Lag_dq = simplify(jacobian(Lag,dq)).';
Lag_dqdq = simplify(jacobian(Lag_dq.',dq));

% The equations have the form W*q_dotdot = RHS, with
W = Lag_dqdq;
RHS = Q + simplify(Lag_q - Lag_qdq*dq);
```

```
state = [q;dq];
param = [J; M; R; g];

matlabFunction(p, 'file','BallPosition','vars',{state,param});
matlabFunction(W,RHS, 'file','BallAndBeamODEMatrices','vars',
{state,To,param});
```

Published with MATLAB® R2019a