```
function E = solveKeplersEq(t, a, e, mu)
% Function that iteratively solves Kepler's Equation for eccentric anomaly
% at some given time past periapsis for a specific 2-body problem orbit
% using Newton's method.
%
%   Inputs:
%       - t: Time past periapsis at which to solve for E, in seconds
%       - a: Semi-major axis of the orbit to analyze, in km
%       - e: Eccentricity of the orbit to analyze
%       - mu: Gravitational parameter of the celestial body for the system
%             to analyze, in km^3/s^2
%   Outputs:
%       - E: Iteratively solved Eccentric anomaly for the given conditions,
%            in rad
%
%   Author: Ian Faber, 09/30/2024
%

% Define Arbitrarily small number for floating point operations to be
% "close enough" to 0
epsilon = 1e-12;

% Define the maximum number of iterations
maxIter = 999;

% Calculate mean anomaly for these conditions
M = sqrt(mu/(a^3))*t; % rad

% Define g functions for Newton's method
g = @(E) E - e*sin(E) - M;
gPrime = @(E) 1 - e*cos(E);

% Iterate E until we converge on a solution
iter = 0; % Initialize at 0 iterations
Ei = M; % Initialize guess at mean anomaly
while iter < maxIter
    E = Ei - (g(Ei)/gPrime(Ei));

    if abs(E - Ei) < epsilon
        iter = iter + 1;
        break; % Stop iteration, we have converged!
    else
        Ei = E;
        iter = iter + 1; % Keep going, still need to converge
    end
end

if iter < maxIter
    fprintf("\nConverged to E = %.6f rad after %.0f iterations!\n", E, iter)
else
    fprintf("\nHit maximum iterations (%.0f) at E = %.6f rad.\n", iter, E)
end
```

```
end
```

```
Not enough input arguments.

Error in solveKeplersEq (line 27)
M = sqrt(mu/(a^3))*t; % rad
```

Published with MATLAB® R2023b