### ODE_Demo_Func.m

```
function [tVec,vVec] = ODE_Demo_Func(vzero,T,epsilon)
% function [tVec,vVec] = ODE_Demo_Func(vzero,T,epsilon)
%
%  Numerically integrates the equation
%
%   dv/dt = -0.2*v(t) + t*sin(t),  v(0) = vzero
%
%  on the interval [0,T], using a time-spacing of epsilon.
%
%  Returns the times in a vector tVec and the corresponding
%  values of v in a vector vVec.

% Generate a vector of epsilon-spaced times from 0 to T.

tVec = [0:epsilon:T];

% The number of values of v that will be computed is the same
% as the number of values of t stored in tVec.

nValues = length(tVec);

% Initialize vVec.  Since tVec(1) = 0, vVec(1) = v(tVec(1)) = v(0) = vzero.

vVec(1) = vzero;

% Iterate through time.

for k = 1:nValues-1
    vVec(k+1) = vVec(k) + epsilon * ( -0.2*vVec(k) + tVec(k)*sin(tVec(k)) );
end

% Return to where the function was called.

return
```

### ODE_Demo.m

```
% Script for calling the function ODE_Demo and plotting the results.
% (Or just type the commands into the MATLAB command window.)

% Numerically integrate dv/dt = -0.2*v(t) + t*sin(t) with v(0) = 5
% over [0,10] using a time-step of .01.
% Call ODE_Demo_Func and assign the results to vectors t and v.

[t,v] = ODE_Demo_Func(5,10,.01);

% Plot the function v versus time t.

plot(t,v);
```

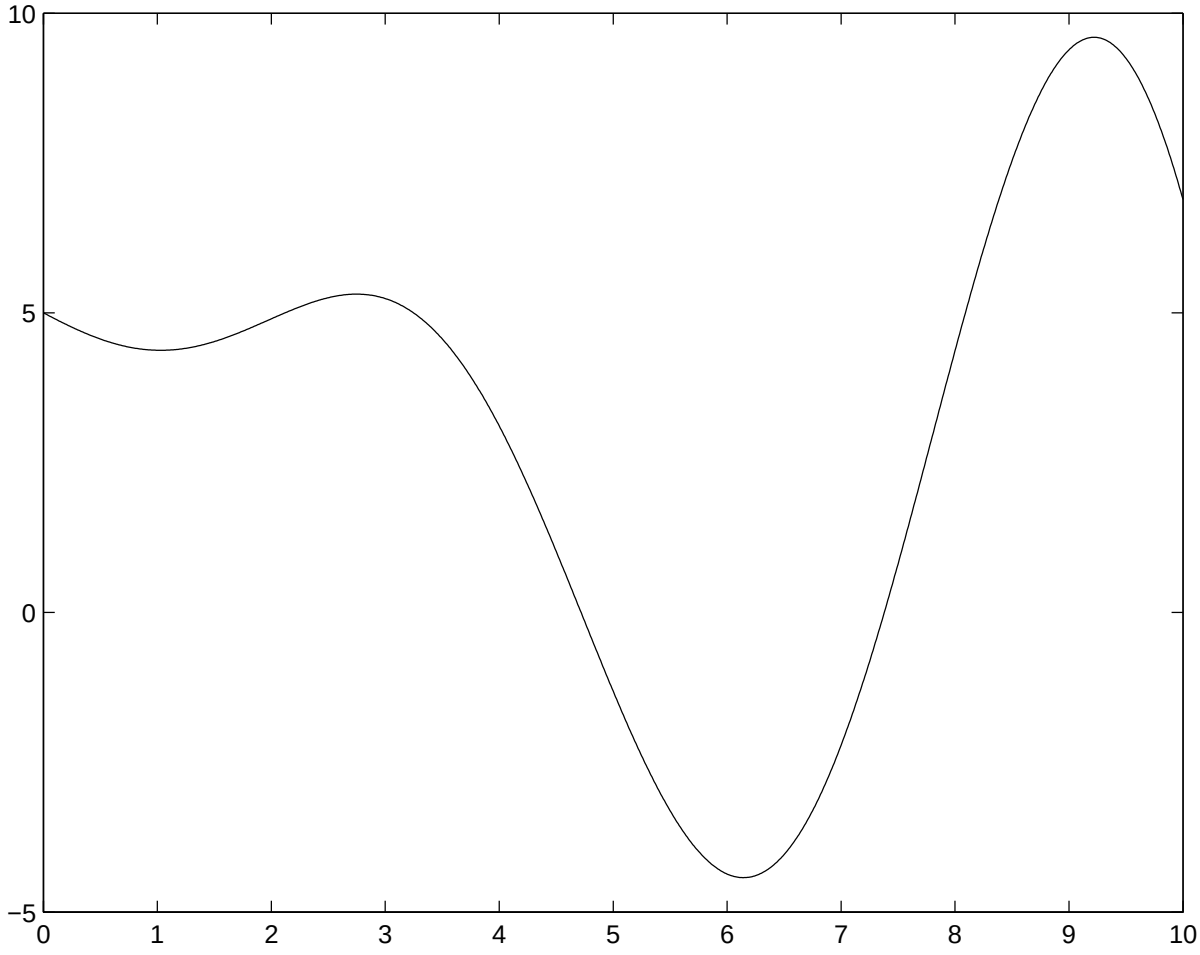


Figure 1: Screen output from `ODE_Demo.m` which uses `ODE_Demo_Func.m`.