```
% advect - program to solve the advection equation
% using the various hyperbolic pde schemes
clear all;  help advect;  % clear memory and print header

%* select numerical parameters (time step, grid spacing, etc.).
method = 1
n = input('enter number of grid points: ');
l = 1.;     % system size
h = l/n;    % grid spacing
c = 1;      % wave speed
fprintf('time for wave to move one grid spacing is %g\n',h/c);
tau = input('enter time step: ');
coeff = -c*tau/(2.*h);  % coefficient used by all schemes
coefflw = 2*coeff^2;    % coefficient used by l-w scheme
fprintf('wave circles system in %g steps\n',l/(c*tau));
nStep = input('enter number of steps: ');

%* set initial and boundary conditions.
sigma = 0.1;              % width of the gaussian pulse
k_wave = pi/sigma;        % wave number of the cosine
x = ((1:n)-1/2)*h - l/2;  % coordinates of grid points
% initial condition is a gaussian-cosine pulse
a = cos(k_wave*x) .* exp(-x.^2/(2*sigma^2));
% use periodic boundary conditions
ip(1:(n-1)) = 2:n;  ip(n) = 1;   % ip = i+1 with periodic b.c.
im(2:n) = 1:(n-1);  im(1) = n;   % im = i-1 with periodic b.c.

%* initialize plotting variables.
iplot = 1;          % plot counter
aplot(:,1) = a(:);  % record the initial state
tplot(1) = 0;       % record the initial time (t=0)
nplots = 50;        % desired number of plots
plotStep = nStep/nplots; % number of steps between plots

%* loop over desired number of steps.
for iStep=1:nStep  %% main loop %%

  %* compute new values of wave amplitude using ftcs,
  %  lax or lax-wendroff method.
  if( method == 1 )      %%% ftcs method %%%
    a(1:n) = a(1:n)*(1+2*coeff) + (-2*coeff*a(im));
  elseif( method == 2 )  %%% lax method %%%
    a(1:n) = .5*(a(ip)+a(im)) + coeff*(a(ip)-a(im));
  else                   %%% Lax-Wendroff method %%%
    a(1:N) = a(1:N) + coeff*(a(ip)-a(im)) + ...
        coefflw*(a(ip)+a(im)-2*a(1:N));
  end

  %* Periodically record a(t) for plotting.
  if( rem(iStep,plotStep) < 1 )  % Every plot_iter steps record
    iplot = iplot+1;
    aplot(:,iplot) = a(:);       % Record a(i) for ploting
    tplot(iplot) = tau*iStep;
    fprintf('%g out of %g steps completed\n',iStep,nStep);
  end
end

%* Plot the initial and final states.
figure(1); clf;  % Clear figure 1 window and bring forward
plot(x,aplot(:,1),'-',x,a,'--');
legend('Initial  ','Final');
xlabel('x');  ylabel('a(x,t)');
pause(1);    % Pause 1 second between plots

%* Plot the wave amplitude versus position and time
figure(2); clf;  % Clear figure 2 window and bring forward
mesh(tplot,x,aplot);
ylabel('Position');  xlabel('Time'); zlabel('Amplitude');
view([-70 50]);  % Better view from this angle
```

```
  advect - program to solve the advection equation
  using the various hyperbolic pde schemes


method =

     1


Error using input
Cannot call INPUT from EVALC.

Error in advect (line 7)
n = input('enter number of grid points: ');
```

Published with MATLAB® R2019a