Problem 2.1 : Matrix creation with loop

i) With nested for loops, control variables $i$ and $j$ indicates row number and column number respectively. $a_{ij} = i + j$ where $i, j = 1, 2, \ldots, 5$

```matlab
>> for i=1:5, for j=1:5, A(i,j)=i+j; end; end;
>> A
A =
     2     3     4     5     6
     3     4     5     6     7
     4     5     6     7     8
     5     6     7     8     9
     6     7     8     9    10
```

ii) Similar to i), $a_{i,j} = (i + j)^2$ where $i, j = 1, 2, \ldots, 5$

```matlab
>> for i=1:5, for j=1:5, B(i,j)=(i+j)^2; end; end;
>> B
B =
     4     9    16    25    36
     9    16    25    36    49
    16    25    36    49    64
    25    36    49    64    81
    36    49    64    81   100
```

Problem 2.2 : Matrix creation with conditional

Mod() function calculates remainder of division. (See mod() in help menu) Unless at least one of a number modulo 2, 3, or 7 is zero, array contains this number.

```matlab
>> C=[]; for i=1:100, ...
```
Problem 2.3 : Velocity and acceleration profile calculation from the ball trajectory

```matlab
function [v1,a1,v2,a2,t]=HW023

% load 'ball.mat'
load ball.mat;

% define time(t) and trajectory(x)
t = A(:,1);  % first column of matrix A
x = A(:,2);  % second column of matrix A

% calculate velocity and acceleration with 'for' loop
% velocity
for i=1:length(x)-1
    v1(i)=(x(i+1)-x(i))/(t(i+1)-t(i));
end;
% acceleration
for i=1:length(v1)-1
    a1(i)=(v1(i+1)-v1(i))/(t(i+1)-t(i));
end;

% calculate velocity and acceleration with 'diff' function
% velocity
v2=diff(x)./diff(t(1:end));
% acceleration
a2=diff(v2)./diff(t(1:end-1));
```

if ~(mod(i,2)==0||mod(i,3)==0||mod(i,7)==0), C=[C i]; end; ...
This neighboring point approach is the best to estimate instantaneous velocity and acceleration of ball if trajectory was measured in the noise free environment. However, in the presence of noise, this approach is bad since it is quite sensitive to noise or disturbance when you estimate instantaneous velocity and even more sensitive in the estimate of instantaneous acceleration due to more noise propagation. To minimize noise sensitivity, averaging data over several data points is recommended such as

\[
v(i) \approx \frac{1}{2} \left( \frac{x(i+1) - x(i)}{t(i+1) - t(i)} + \frac{x(i) - x(i-1)}{t(i) - t(i-1)} \right)
\]