```
% J Hundley
% assign03.m
% find the gravitation force from between two objects

clc, clear all
format long
format compact

 ***** CONSTANTS *****
GRAVITY = 6.67384e-11;  % Newtons
MINMASS = 1000;         % kg
MAXMASS = 100000;       % kg
MINDIST = 1000;         % m
MAXDIST = 100000;       % m

% ***** INPUT *****
% get matrix of masses
randNum = rand(5,2);
mass    = MINMASS + (MAXMASS-MINMASS) .* randNum; % kg
% get vector of distances
randNum = rand(5,1);
dist    = MINDIST + (MAXDIST-MINDIST) .* randNum; % m

% ***** COMPUTE *****
% compute vector forces between mass-1 and mass-2
force = GRAVITY .* mass(:,1) .* mass(:,2) ./ dist .^2; % Newtons
% create table: mass, distant, force
table = [ mass, dist, force ];

% get maximum force from table and the row number
[ maxForce, rowNum] = max( table(:,4) );

% compute force using max masses and distance
newForce = GRAVITY .* max(mass(:,1)) .* max(mass(:,2)) ./ max(dist) .^2; % Newtons

% ***** OUTPUT *****
% display table with title and column headers
disp( '   Gravitation force table (Newtons):' )
disp( '   Mass-1 (kg)         Mass-2 (kg)         Distant (m)         Force (N)' )
disp( table )
disp(' ')

% display maximum force and row number
disp( [ 'Max force: ', num2str(maxForce), ' (N) on row ', num2str(rowNum) ] )
disp(' ')

% display values used to compute force from maximum column values
disp( '   Gravitation force from min masses and distance (Newtons):' )
disp( '   Mass-1 (kg)         Mass-2 (kg)         Distant (m)         Force (N)' )
disp( [ max(mass(:,1)), max(mass(:,2)), max(dist), newForce ] )
disp(' ')

% display table with title and column headers sorted by force descended
disp( '   Gravitation force table sorted by force (Newtons):' )
disp( '   Mass-1 (kg)         Mass-2 (kg)         Distant (m)         Force (N)' )
disp( sortrows( table, -4 ) )
```

Use colon notation with columns for masses NOT vectors.

Use nested max() NOT variable.

*Read **all** instructions*
before beginning your work.

COMP1200-MATLAB - assign 03
Due 4:45pm – Friday – September 20, 2019
Submit `assign03.m` via Canvas

NOTE:
Your submitted file(s) MUST be spelled and cased as instructed.

***Before you start writing your program*:**
Read the complete instructions including the **algorithm**. An **algorithm** contains the steps needed to guide you through solving a problem. Use the **algorithm** as comments to a guide you when writing the MATLAB program file solution for the following problem.

***Problem*:**
The Force of Gravity equation computes the gravitational force between the two masses: the Sun and the planet or other body from our solar system. In this equation, the masses are treated as point masses separated by a specified distance. The Universal Gravitational Constant is 6.67384E-11 (N).

$$F = G\frac{m_1 m_2}{r^2}$$

where:

- F is the force between the masses (Newtons),
- G is the gravitational constant (6.673×10^{-11} N),
- m_1 is the first mass (kilograms),
- m_2 is the second mass (kilograms), and
- r is the distance between the centers of the masses (meters).

Instructions for all assignment scripts:

- □ See Standards for Documentation of MATLAB Programs on the Canvas Resources page.
- □ Insert comments at the top and throughout each file.
  - o Include the follow comments at the beginning of this (and ALL) files.
    - % submitter's name, **GROUP # or "none"**
    - % other group members' names or "none"
    - % **`program file name`**, ex. assign02a.m
    - % due date of the assignment
    - % **statement about collaboration REQUIRED.**
    - % a short narrative about what the file does
  - o Use the algorithm given as comments throughout your program.
- □ Observe the instructor's rule for naming variables.
  - o Use ALL CAPS for constants variable names.
  - o Start other variables with lower case.
  - o Use descriptive variable names.
- □ Use Sample Input/Output as a guide.
- □ Code clarity:
  - o Indent blocks as needed. **Use Smart Indent.**
  - o Divide your solution program code into sections as noted in the algorithm. Use blank lines as needed to group statements.
  - o Use section comments as well as the algorithm step comments.
  - o Remove statements from previous assignments that do not apply to the current requirements.
- □ Use comments to show units.
- □ **Use the CONSTANT and variable names, not numbers.**
  Exceptions are incrementers (or counters) and numbers without identity.
- □ No extra output, i.e. use semicolons

GRADE OF ZERO for a file if submitter name not part of Canvas group.

(-3pts) No CURRENT GROUP# or "none".
(-3pts) For your own protection, type "none" for other group members if submitting alone.
(-5pts) Five point penalty for not joining your Canvas group.

(-5pts) Zero points for comments if no collaboration statement.

NOTE – There can be multiple ways to solve a problem with software.
The assignment instructions are developed to introduce you to specific skills and functions.
Following the instructions is a significant part of the learning process and grading.
Use skills designated in the green box. Do not use skills not covered to this point.

***Program*: assign03.m** **(Detailed instructions for algorithm.)**
You are to write a program (a MATLAB script file) that uses a matrix of random masses and a vector of distances to compute and display the forces.

The following will provide information needed to accomplish the steps in the algorithm below.

- get matrix of masses
  Create a 5x2 matrix of random numbers (masses) in the **given range**. Col-1 is mass-1. Col-2 is mass-2.
- get vector of distances
  Create a 5x1 vector of random numbers (distances) in the **given range**.
- compute vector forces between mass-1 and mass-2 with given distances
  Use colon notation to use the columns of the mass matrix for mass-1 and mass-2.
- create table from input and output: mass (col-1,2), distant (col-3), force (col-4)
  See the sample output.
- get maximum force from table and the row number
  Find the maximum **force*** (col-4) and its **row.****
- compute force using max masses and distance
  Compute the **force***** for the **maximum mass-1, mass-2, and distance columns**.
  Use the `max()` function with appropriate columns in the equation instead of variables.
- display table with title and column headers
  Display a blank line after the table.
- display maximum force and row number
  Using one display statement, display the maximum **force*** and **row number**** on one line with labels.
- display values used to compute force from maximum column values
  Display the **maximum values from columns 1-3** and the computed **force***** with same title and column headings as table.
  Use the `max()` function with appropriate columns in the equation instead of variables as above.
- display table with title and column headers sorted by force descended
  Display the table sorted by the force column in descending order. Nest the `sortrows()` function as input for the `disp()`.

*, **, *** help to link information in multiple statements.

***Problem* CONSTANTS:** (with units) **Use CONSTANT names not numbers.**

```
   GRAVITY 6.67384e-11 % Newtons
   MINMASS 1000        % kg
   MAXMASS 100000      % kg
   MINDIST 1000        % m
   MAXDIST 100000      % m
```

***Problem Inputs*:** (with units)

5x2 matrix of random mass between min & max mass `% kg`
5x1 vector of random distance between min & max distances `% m`

Problem Outputs: (with units)

Vector of force `% kg`

Other variables: (with units)

As needed in instructions

Equation:

See above.

Algorithm: See **green** comments below.

Type the comments in the editor window and
use as a guide for typing the MATLAB statements.

Start your program file by typing the following into your empty editor window.

- Type yours/your group and other required information comments.
- Type the algorithm as given below as comments to guide you when writing the MATLAB instructions to do the tasks to solve the given problem.
- Below the comment, type the MATLAB statement(s) that do what the comment says. This example should help. →

**NO error checking.
Do not use commands and statements beyond what has been taught on class.**

New commands:
`rand()` see p.99 in text
`max()` with one & two output
Use colon notation for value in matrix and not variables.
Create table from matrix and vectors.
Display a vector of values.
`num2str()`
`sortrows()` descending

Continue to use:
`format long, compact`
`disp()`
Use descriptive variables.

% submitter's name, **GROUP # or none**
% other group members' names **or none**
% **`program file name`**
% due date of the assignment
% **statement(s) about collaboration. See syllabus for examples.**
% a short narrative about what the file does

```
clc, clear all
format long
format compact
% ***** CONSTANTS *****

% ***** INPUT *****
% get matrix of masses
% get vector of distances

% ***** COMPUTE *****
% compute vector forces between mass-1 and mass-2
% create table: mass, distant, force
% get maximum force from table and the row number
% compute force using max masses and distance

% ***** OUTPUT *****
% display table with title and column headers
% display maximum force and row number
% display values used to compute force from maximum column values
% display table with title and column headers sorted by force descended
```

NOTE: Your submitted file(s) MUST be spelled and cased as instructed.

One submission per group. Canvas links members to files and rubric.

Submit via Canvas:
`assign03.m` MATLAB script file

Sample output: **Each run will have different numbers.**

```
ONE SAMPLE RUN =================================================================

Gravitation force table (Newtons):
   Mass-1 (kg)          Mass-2 (kg)          Distant (m)          Force (N)
   1.0e+04 *
   8.165764495292471    1.065650009494154    1.660369508607728    0.000000000000021
   9.067340177048631    2.857132366783779    9.708868539430094    0.000000000000002
   1.357169481305710    5.514127040129340    9.575952787605161    0.000000000000001
   9.142420975776291    9.579317670799545    4.905218922356128    0.000000000000024
   6.360356537631554    9.652396498472838    8.022776641999121    0.000000000000006

Max force: 2.4292e-10 (N) on row 4

Gravitation force from max masses and distance (Newtons):
   Mass-1 (kg)          Mass-2 (kg)          Distance (m)         Force (N)
   1.0e+04 *
   9.142420975776291    9.652396498472838    9.708868539430094    0.000000000000006

Gravitation force table sorted by force (Newtons):
   Mass-1 (kg)          Mass-2 (kg)          Distance (m)         Force (N)
   1.0e+04 *
   9.142420975776291    9.579317670799545    4.905218922356128    0.000000000000024
   8.165764495292471    1.065650009494154    1.660369508607728    0.000000000000021
   6.360356537631554    9.652396498472838    8.022776641999121    0.000000000000006
   9.067340177048631    2.857132366783779    9.708868539430094    0.000000000000002
   1.357169481305710    5.514127040129340    9.575952787605161    0.000000000000001

ANOTHER SAMPLE RUN =============================================================

Gravitation force table (Newtons):
   Mass-1 (kg)          Mass-2 (kg)          Distance (m)         Force (N)
   1.0e+04 *
   1.504674752409432    6.591832921650210    7.601627292725500    0.000000000000001
   4.275436698000122    0.453545617884477    7.457011434436669    0.000000000000000
   9.165781699371765    8.506380128100894    3.983047493388265    0.000000000000033
   7.942852562639588    9.346533152799751    6.589231112757811    0.000000000000011
   9.598975021289741    6.819478033091958    1.794748209334461    0.000000000000136

Max force: 1.3563e-09 (N) on row 5

Gravitation force from max masses and distance (Newtons):
   Mass-1 (kg)          Mass-2 (kg)          Distance (m)         Force (N)
   1.0e+04 *
   9.598975021289741    9.346533152799751    7.601627292725500    0.000000000000010

Gravitation force table sorted by force (Newtons):
   Mass-1 (kg)          Mass-2 (kg)          Distance (m)         Force (N)
   1.0e+04 *
   9.598975021289741    6.819478033091958    1.794748209334461    0.000000000000136
   9.165781699371765    8.506380128100894    3.983047493388265    0.000000000000033
   7.942852562639588    9.346533152799751    6.589231112757811    0.000000000000011
   1.504674752409432    6.591832921650210    7.601627292725500    0.000000000000001
   4.275436698000122    0.453545617884477    7.457011434436669    0.000000000000000
```