```
% J Hundley   % assign04a.m
% find the gravitation force from sun to a specific planet
clc, clear all
format long
format compact

% ***** CONSTANTS *****
GRAVITY  = 6.67384e-11;       % Newtons
MASS_SUN = 1.989e30;         % kg

% ***** INPUT *****
% get vectors of masses, distances for planets
planetMass = [ 0.330, 4.87, 5.97, 0.642, 1898, 568, 86.8, 102 ] * 10^24;          % kg
planetDist = [ 57.9, 108.2, 149.6, 227.9, 778.6, 1433.5, 2872.5, 4495.1 ] * 10^9; % m

% ask user for planet number within range
planetNum = 0;
while planetNum < 1 || planetNum > 8
    planetNum = input( 'Enter a planet number (1-8): ' );
end

% ***** COMPUTE *****
% compute force from sun to selected planet
force = GRAVITY * planetMass(planetNum) .* MASS_SUN ./ planetDist(planetNum) .^2; % Newtons,
scalars dot optional

% ***** OUTPUT *****
% display information with title and headers
disp( [ '   Gravitation force from sun to planet #' num2str(planetNum) ] )
disp( '   Mass (kg)           Distant (m)         Force (N)' )
disp( [planetMass(planetNum) planetDist(planetNum) force ] )
```

```
% J Hundley   % assign04b.m
% find the gravitation force from sun to planets
clc, clear all
format long
format compact

% ***** CONSTANTS *****
GRAVITY = 6.67384e-11;       % Newtons
MASS_SUN = 1.989e30;         % kg

% ***** INPUT *****
% get vectors of masses, distances, given forces for planets
planetMass =  [ 0.330, 4.87, 5.97, 0.642, 1898, 568, 86.8, 102 ] * 10^24;          % kg
planetDist =  [ 57.9, 108.2, 149.6, 227.9, 778.6, 1433.5, 2872.5, 4495.1 ] * 10^9; % m
planetForceGiven = [ 1.30637878E+22, 5.53822703E+22, 3.54146321E+22, 1.63899504E+21,...
    4.16149445E+23, 3.68628209E+22, 1.39851857E+21, 6.6847851E+20 ];               % N

% select a planet from menu
planetNum = menu( 'Select a
planet','MERCURY','VENUS','EARTH','MARS','JUPITER','SATURN','URANUS','NEPTUNE' );

% ***** COMPUTE *****
% compute force from sun to planet
compForce = GRAVITY * planetMass(planetNum) .* MASS_SUN ./ planetDist(planetNum) .^2; % Newtons

% ***** OUTPUT *****
% display information with title and headers
disp( [ '   Gravitation force from sun to planet #' num2str(planetNum) ] )
disp( '   Mass (kg)           Distant (m)         Computed Force (N)  Given Force (N)' )
disp( [planetMass(planetNum) planetDist(planetNum) compForce planetForceGiven(planetNum) ] )
% print message noting whether computed force is > or < than given force
if compForce < planetForceGiven(planetNum)
    disp( 'computed force < given force' )
else
    disp( 'computed force > given force' )
end
```

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

COMP1200-MATLAB - assign 04
Due 4:45pm – Friday – September 27, 2019
Submit `assign04a.m and assign04b.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 m^3 / kg * sec^2.

where:

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

- F is the force between the masses (Newtons),
- G is the gravitational constant (6.673×10^{-11} N·$(m/kg)^2$),
- 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).

	MERCURY	VENUS	EARTH	MARS	JUPITER	SATURN	URANUS	NEPTUNE
Mass (10^{24}kg)	0.330	4.87	5.97	0.642	1898	568	86.8	102
Distance from Sun (10^6 km)	57.9	108.2	149.6	227.9	778.6	1433.5	2872.5	4495.1
Force of sun's gravity (N)	1.30637878 E+22	5.53822703 E+22	3.54146321 E+22	1.63899504 E+21	4.16149445 E+23	3.68628209 E+22	1.39851857 E+21	6.6847851 E+20

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.

***Program*: assign04a.m**
Write a program (a MATLAB script file) that uses the vector of planet masses, mass of Sun, and vector of distances from Sun to planets. Compute and display the force of the Sun to a user selected planet with appropriate labeling.

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

```
gravity      6.67384e-11 % Newtons
mass of sun  1.989e30    % kg
```

***Problem Inputs*:** (with units)
vector of planet masses % kg
vector of planet distances % m
Problem Outputs: (with units)
Force % N
Other variables: (with units)
planet number
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. →

```
% 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 vectors of masses, distances for planets

% ask user for planet number within range

% ***** COMPUTE *****
% compute force from sun to selected planet

% ***** OUTPUT *****
% display information with title and headers
```

Do not use commands and statements beyond what has been taught on class.

New commands:
`input()`
`while` data validation loop
relational, logical operators
indexing

Continue to use:
`format long, compact`
`disp()`
`num2str()`
Use descriptive variables.

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

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

Sample output: **Test with good and bad responses.**

```
Enter a planet number (1-8): 0
Enter a planet number (1-8): 9
Enter a planet number (1-8): 5
   Gravitation force from sun to planet #5
   Mass (kg)           Distant (m)         Force (N)
   1.0e+27 *
   1.898000000000000   0.000000000000001   0.000415602338942
```

***Program*: assign04b.m**
Write a program (a MATLAB script file) that uses the matrix of planet masses, mass of Sun, vector of distances from Sun to planets, and vector of given forces. Compute and display the force of the Sun to a user selected planet with appropriate labeling. Print a note comparing given and computed forces. The planet names are needed with menu MERCURY, VENUS, EARTH, MARS, JUPITER, SATURN, URANUS, NEPTUNE.

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

```
    gravity      6.67384e-11 % Newtons
    mass of sun  1.989e30    % kg
```

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

```
    vector of planet masses      % kg
    vector of planet distances   % m
    vector of given forces       % N
```

Problem Outputs: (with units)

```
    force                        % N
```

Other variables: (with units)
planet number

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. →

Do not use commands and statements beyond what has been taught on class.

New commands:
`menu()`
`if_else` to compare given & computed force
no `if_elseif_else` no used here
relational operators
indexing

Continue to use:
`format long, compact`
`disp()`
`num2str()`
Use descriptive variables.

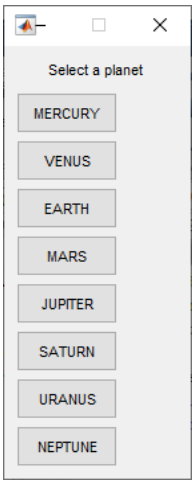
NOTE: Your submitted file(s) MUST be spelled and cased as instructed.

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

Submit via Canvas:

```
    assign04a.m   MATLAB script file
    assign04b.m   MATLAB script file
```

```
% 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 vectors of masses, distances, given forces for planets

% select a planet from menu

% ***** COMPUTE *****
% compute force from sun to selected planet

% ***** OUTPUT *****
% display information with title and headers

% print message noting whether computed force is > or < than given force
```



Sample output: **Test with both < > comparison responses.**

```
   Gravitation force from sun to planet #4
   Mass (kg)          Distant (m)         Computed Force (N)  Given Force (N)
   1.0e+23 *
   6.420000000000001   0.000000000002279   0.016408049268201   0.016389950400000
computed force > given force
```