Name: _____

Date: _____

Collaborators: _____

(If applicable, the collaborators submit their individually written assignments together)

Question:	1	2	3	4	5	Total
Points:	5	25	20	15	15	80
Score:						

Instructor/grader comments:

1. Project initialization:

   (a) (0 points) ssh to your virtual machine (also creating an ssh tunnel that connects ports 8888 on both ends of the tunnel).

   (b) (0 points) Once per assignment: create a directory for this homework assignment (say **hw07**)

   (c) (5 points) Once per assignment: from the command prompt start julia, switch to the package mode and activate your project. Install the package `IJulia` locally - that is needed to create Project.toml and Manifest.toml project files. (You can add other required packages when working with notebooks.)

   (d) (0 points) Exit julia and launch a jupyter server on the virtual machine, e.g.

   ```
   julia -e 'using IJulia; notebook(dir=".",detached=true)'
   ```

2. Working with observational data (aka "big data")

   (a) (5 points) Download the catalog of white dwarfs observational data from The VizieR database of astronomical catalogues and clean the data. Use the following fragment of the code:

   ```
   url = "http://vizier.u-strasbg.fr/viz-bin/asu-tsv?-source=J/A+A/420/5
   catalog = download(url)
   df = CSV.read(catalog, DataFrame, comment="#", skipto=4,
                 types=[Float64, Float64], delim='\t',
                 ignorerepeated=true, silencewarnings=true)
   dropmissing!(df)
   ```

   The complete code to start that part of the assignment is available on the class website at the file named **hw07p2.ipynb**

   (b) (10 points) Use the function `linear_regression` that you wrote for Midterm 2 to find the best linear fit the data. On the same figure plot the data and the fit. Provide axes labels, grid, legend, title.

   (c) (10 points) Use the parameters of your fit to approximately determine (and print) the Chandrasekhar limit for white dwarfs. In your README.md file discuss how close the result of using the linear fit is to the result of a better theory (discussed in class).

3. Bouncing ball on an oscillating table

Consider a ball bouncing vertically on an oscillating table. The position of the table is a prescribed function of time,

$$y(t) = a\sin(\omega t), \tag{1}$$

where a is the amplitude of the oscillations, ω is its frequency.

The motion of the ball between bounces is governed purely by gravity:

$$\frac{\mathrm{d}^2 y}{\mathrm{d}t^2} = -g, \tag{2}$$

where g is the acceleration of gravity.

The interaction of the ball and the table occurs only at the bounces, which assumed to be elastic. That is, the velocities of the ball immediately before and after the collision, v_b and v_a, are related as following:

$$v_a = -v_b + 2v_t, \tag{3}$$

where v_t is the velocity of the table at the instance of the collision.

For the numerical calculations below, use the following parameters: $g = 10\mathrm{m/s}^2$, $a = 1\mathrm{m}$, $\omega = 10\ \mathrm{sec}^{-1}$, $y(0) = 50\mathrm{m}$, $\dot{y}(0) = 0$.

Create a separate jupyter notebook (say **hw07p3.ipynb**) for the problem.

(a) (10 points) Rewrite the equations om motion Eq. (1) and (2) in the dimensionless form: measure distance in the units of a,

$$y = au; \tag{4}$$

measure time in the units of ω^{-1},

$$t = \frac{\tau}{\omega}. \tag{5}$$

Notice that dimensionless form of the equations contains only one parameter (vs three in the dimensional form).

Show your calculations and the result in the space below:

(b) (10 points) Solve the dimensionless equation numerically. Use the event handling machinery provided by the differential equation solver. Follow the motion of the ball for the time span `tspan = (0., 10000.)`.

Use the following code fragments to handle the bounces:

```
function condition(u, t, integrator)
  u[1] - sin(t)
end

function affect!(integrator)
  integrator.u[2] = -integrator.u[2] + 2*cos(integrator.t)
end
```

Provide an extra parameter, `dtmax=0.1`, for the differential equation solver:

```
sol = solve(prob, Tsit5(), callback=cb, dtmax=0.1)
```

Plot the results of your calculations as following:

```
plot(sol.t, sol[1,:])
```

In your README.md file state whether the ball lost or gain energy.

Git

4. (15 points)

Clean the cells of your jupyter notebook(s).

Create a git repository for hw07. Check your notebook(s), Julia project files (Project.toml and Manifest.toml) , .gitignore file into the repository. Provide meaningful commit messages.

Gitlab

5. (15 points)

   Create an empty Gitlab project called **hw07** (name it exactly as shown). At this step un-check the box "Create README.md file".

   Push the content of your git repository to Gitlab's **hw07** project

   On the Gitlab "side" create README.md file.

   Pull the README.md file to your local git repository to synchronize your local and remote repositories.

   I have synchronized the contents of my local and remote repositories that I created for hw07 assignment

   Sign and date here: _____

   Share the project with the instructor and grant him **Reporter** privileges.