N-Body Simulation

CSE1010 Project 6, Fall 2012

Date: 10/22/2012

Name: William Dickson

Section: 009L

TA: Levon Nazaryan

Instructor: Jeffrey A. Meunier

1. **Introduction**

In this project you will write a program that calculates and plots the trajectory of a projectile. The user is prompted to enter values for mass, energy, angle, and the initial height of the projectile, and the program will calculate and plot the projectile's trajectory. This simulation will ignore complicating factors such as friction, air resistance, spin, and rebound. A trajectory is the path followed by a projectile. A projectile is an unsupported object that is moving through space under some force (like a rocket) or under its own momentum (like a ball or a rock).

1. **Test runs**
   1. **Test run of mass 50, energy .5, angle 45, and starting height .1**

Enter a negative mass to exit the program.

Enter a mass in grams: 50

Enter an energy in Joules: .5

Enter an angle in degrees: 45

Enter a starting height in meters: .1

Mass = 0.05 kg

Energy = 0.5 J

Angle = 0.785398 rad

Starting height (y0) = 0.1 m

Velocity = 4.47214 m/s

Vx = 3.16228 m/s

Vy = 3.16228 m/s

Time to apogee = 0.322681 s

Height at apogee = 0.610204 m

Time to fall to earth = 0.35289 s

Total time = 0.675572 s

Distance traveled = 2.13634 m

**D:\Users\Liam\Documents\Dropbox\UCONN\Freshman\CSE1010\HW05\test1.tif**

* 1. **Test run of mass 50, energy 2, angle 10, and starting height 0**

Enter a negative mass to exit the program.

Enter a mass in grams: 50

Enter an energy in Joules: 2

Enter an angle in degrees: 10

Enter a starting height in meters: 0

Mass = 0.05 kg

Energy = 2 J

Angle = 0.174533 rad

Starting height (y0) = 0 m

Velocity = 8.94427 m/s

Vx = 8.80839 m/s

Vy = 1.55316 m/s

Time to apogee = 0.158485 s

Height at apogee = 0.123076 m

Time to fall to earth = 0.158485 s

Total time = 0.316971 s

Distance traveled = 2.792 m

D:\Users\Liam\Documents\Dropbox\UCONN\Freshman\CSE1010\HW05\test2.tif

* 1. **Test run of mass 50, energy .5, angle 45, and starting height .1**

Enter a negative mass to exit the program.

Enter a mass in grams: 10

Enter an energy in Joules: 4

Enter an angle in degrees: 70

Enter a starting height in meters: 0

Mass = 0.01 kg

Energy = 4 J

Angle = 1.22173 rad

Starting height (y0) = 0 m

Velocity = 28.2843 m/s

Vx = 9.67379 m/s

Vy = 26.5785 m/s

Time to apogee = 2.71209 s

Height at apogee = 36.0417 m

Time to fall to earth = 2.71209 s

Total time = 5.42419 s

Distance traveled = 52.4725 m

D:\Users\Liam\Documents\Dropbox\UCONN\Freshman\CSE1010\HW05\test3.tif

1. **Source Code**

% Trajectories

% CSE1010 Project 5, Fall 2012

% Name: William Dickson

% Date: 10/10/2012

% Section: 009L

% TA: Levon Nazaryan

% Instructor: Jeffrey A. Meunier

clc % clear command window

clear % clear all variables

clf % clear plot area

while true

setupGlobals

i = getInputs;

if isempty(i) == 1

break

end

ci = convertInputs(i);

displayConvertedInputs(ci);

d = calcDependents(ci);

displayDependents(d);

t = calcTrajectory(ci,d);

plot(t(1,1:end),t(2,1:end),'o')

end

function d = calcDependents(ci)

% Calculates the dependents given the converted inputs. Returns the

% values for velocity, x velocity, initial y velocity, time to apogee,

% the y coordinate of apogee, the time to fall, the total time, and

% horizontal distance in a vector (in that order).

% Use: calcDependents(ci)

global G

m = ci(1);

e = ci(2);

a = ci(3);

y0 = ci(4);

v = sqrt(2\*e/m);

vx = v\*cos(a);

vy = v\*sin(a);

tap = -vy/G;

yap = y0 + vy\*tap + .5\*G\*tap^2;

tfall = sqrt(-2\*yap/G);

t = tap + tfall;

dh = vx\*t;

d = [v vx vy tap yap tfall t dh];

end

function t = calcTrajectory(ci,d)

% Calculates the trajectory of a particle given the converted inputs and

% the dependents. Returns the path of the trajectory as (x,y) coordinates

% with the x coordinates in the first row of the return vector and the y

% coordinates in the second row.

% Use: calcTrajectory(ci,d)

global G

ts = 0:.01:d(7);

xs = linspace(0,d(8),length(ts));

ys = ci(4) + d(3).\*ts + .5\*G.\*ts.^2;

t = [xs;ys];

end

function ci = convertInputs(i)

% Used for converting element 1 of the vector "i" from g to kg and

% element 3 from degrees to radians. Returns the original vector with

% elements 1 and 3 converted.

% Use: convertInputs(i)

i(1) = i(1)/1000;

i(3) = i(3)/180\*pi;

ci = i;

end

function displayConvertedInputs(ci)

% Displays the "converted inputs" or any values stored in the first 4

% elements of a vector) in order.

% Use: displayConvertedInputs(ci)

fprintf('Mass = %g kg\n',ci(1))

fprintf('Energy = %g J\n',ci(2))

fprintf('Angle = %g rad\n',ci(3))

fprintf('Starting height (y0) = %g m\n',ci(4))

end

function displayDependents(d)

% Displays the "dependents" (or any values stored in the first 8

% elements of a vector) in order.

% Use: displayDependents(d)

fprintf('Velocity = %g m/s\n',d(1))

fprintf('Vx = %g m/s\n',d(2))

fprintf('Vy = %g m/s\n',d(3))

fprintf('Time to apogee = %g s\n',d(4))

fprintf('Height at apogee = %g m\n',d(5))

fprintf('Time to fall to earth = %g s\n',d(6))

fprintf('Total time = %g s\n',d(7))

fprintf('Distance traveled = %g m\n',d(8))

end

function i = getInputs

% Prompts the user for input values for mass, energy, angle and starting

% height which is then returned in a 1x4 vector

% Use: getInputs

m = input('Enter a negative mass to exit the program.\nEnter a mass in grams: ');

if m >= 0

e = input('Enter an energy in Joules: ');

a = input('Enter an angle in degrees: ');

h = input('Enter a starting height in meters: ');

i = [m e a h];

else

i = [];

end

end

function setupGlobals

% Defines G to be a global variable and give it a value:

% Use: setupGlobals

global G

G = -9.8;

end