CSCE 1030 – Homework 1

Due: 11:59 PM on Wednesday, September 17, 2014 CST

Problem Statement:

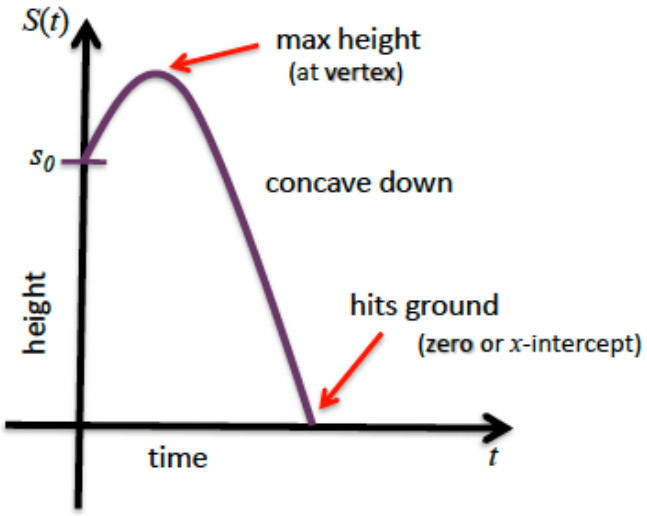
The purpose of this programming project is to write a small C program to calculate the effect of gravity on a projectile, that is, an object that is thrown, shot, or dropped. More specifically, the height of an object propelled directly upward from an initial height s_0 at an initial velocity v_0 is given by

$$S(t) = -16t^2 + v_0 t + s_0,$$

where v_0 is in feet/second and s_0 is in feet. The maximum height (at the vertex) can be found at the coordinates:

$$\left(\frac{-b}{2a}, f\left(\frac{-b}{2a}\right)\right) = \left(\frac{-b}{2a}, \frac{-b^2 + 4ac}{4a}\right)$$

Visually, you may view the height of an object over time by the following graph:



Using the quadratic equation $x = \dfrac{-b \pm \sqrt{b^2 - 4ac}}{2a}$, you are to calculate how long it takes the projectile to hit the ground, as defined by the zero, or x-intercept, of the equation (see above diagram). Additionally, you are to calculate the maximum height of the object attains using the formula for the coordinates given above. You are to prompt the user for the initial height s_0 and the initial velocity v_0 as floating-point numbers, calculate the number of seconds the projectile takes to hit the ground rounded to two decimal places, calculate the object's maximum height rounded to two decimal places, and then print those results to the screen. You may assume that all input will be of the appropriate data type, so you do not need to do error checking. You do not need to implement a loop to read the number; just use read statements. Note, however, that will possibly have two solutions (i.e., x-intercepts) to the number of seconds it takes for the object to hit the ground, but you are only to print the valid solution.

CSCE 1030 – Homework 1

Due: 11:59 PM on Wednesday, September 17, 2014 CST

Design:

On a piece of paper, use the formulas given above (and a calculator) to work out what the answer should be for one or two sets of inputs.

Write down in English the sequence of steps you performed above. Pretend this is a "recipe" for someone else to follow. Now, follow your instructions and see if you can calculate the results for several different sets of inputs. Refine your "recipe" until it is clear. Be sure to include the steps for prompting for input, etc.

Type these steps into a document (Word, txt, PDF, etc.). Also be sure to include your algorithm steps as comments in your code file.

Implementation:

Now that you have a working design, your next step is to translate these steps into C code. Use the algorithm development techniques discussed in class to implement your solution to the problem above. Add your C code a little at a time, and compile and test as you go.

Remember to add your comments to your code to explain your program. Do this before/during programming instead of waiting until the end. At a minimum, you should comment the header (e.g., name, class, date, brief description of the program, etc.), all variables (i.e., what they are used for), and specific "blocks" of code. For example, use comments to describe the inputs, the formulas used, and any other important steps in your code.

Your program will be graded based largely upon whether it works correctly on a CSE Department machine, so you should make sure your program compiles and runs on a CSE machine.

Your program will also be graded based upon your programming style. At the very least, your program should include:

- A consistent indentation style as recommended in the textbook and in class;
- Meaningful variable names;
- A block header comment section that includes: your name, e-mail address, and a brief description of the program.

Your program's output should initially display the department and course number, program number, your name, your EUID, and your e-mail address.

CSCE 1030 – Homework 1

Due: 11:59 PM on Wednesday, September 17, 2014 CST

Testing:

Test your program to check that it operates as desired with a variety of inputs. Compare the answers your code gives with the ones you get from hand-calculations. Although your program is not required to check for incorrect inputs, observe the effect of such inputs. Try typing "hello world" when your program asks for a number. What is the result?

Here is an example of a sample run: (input is in **bold green**)

```
Enter initial height of s0 in feet: 80
Enter initial velocity of v0 in feet/sec: 32
The projectile will hit the ground in 3.45 seconds.
The maximum height the object will reach is 96.00 feet.
```

Documentation:

When you have completed your C program, write a short report (2 – 3 paragraphs) describing what the objectives were, what you did, and the status of the program. Does it work properly for all test cases? Are there any known problems? Save this report in a separate file to be submitted electronically.

Homework Submission:

In this class, we will be using electronic homework submission to make sure that all students hand their programming projects (and labs) on time. You will submit your program source file to the class website through the "**Homework 1**" drop box by the due date and time.

Note that this project must be done individually. The program will be checked using a code plagiarism tool against other solutions, so please ensure that all work submitted is your own.

Note that the dates on your electronic submission will be used to verify that you met the due date above. All homework up to 24 hours late will receive a 50% grade penalty. Later submissions will receive zero credit, so hand in your best effort on the due date.

Summary:

- You will design a solution to the problem.
- You will implement it on the CSE machines using C. You will make sure to use good style, good variable names, indentation, etc. You will compile, run, and test your code.
- You will write a brief report describing what your code does and how well it works.
- You will submit electronically your C code, your design, and your brief report.