Week 4 Lab Part 2

1. Write a program that will let the user enter a grade, and the program will output what number they should expect with that grade. e.g. If I were to enter C-, the computer will tell me “1.7”. Use the following list as a grading scale. Assume that the user will only enter string ranging from B- to A+. You must use nested statements and the input must be a single string, not two CHAR nor two strings.

You can NOT use (if string\_from\_user == “A+”). (Syntactically you can, but for this problem you cannot.)

Use this grading scale:

A+ = 4.0

A = 4.0

A- = 3.7

B+ = 3.3

B = 3.0

B- = 2.7

1. The Babylonian algorithm to compute the square root ***n*** is as follows:
   1. Make a ***guess*** at the answer (you can pick ***n / 2*** as your initial guess)
   2. Compute ***r = n / guess***
   3. Set ***guess = (guess + r) / 2***
   4. Go back to step 2 for as many iterations as necessary. ***Guess*** will be become closer to the square root as the steps are repeated.

Write a program that inputs an integer for ***n***, iterates through the Babylonian algorithm until ***guess*** is within 1% of the previous guess. Output the answer as a double with 5 decimals.

1. A year with 366 days is called a leap year. A year is a leap year if it is divisible by 4 (for example, 1980), except that it is not a leap year if it is divisible by 100 (for example, 1900). However, a leap year is divisible by 400, (for example, 2000). Write a program that asks the user for a year and computes whether that year is a leap year.
2. Do problem 3 with only one compounded if statement.
3. Create a guessing game. The computer randomly selects a number between 1 and 100. If the user enters a number lower than the selected number, tell the user they have selected a too small number. If the user enters a larger number, tell the user they entered a number too large. Allow the user only 10 guesses before stating game over.