Assignment #5
Designing a Computational Solution
Due: Monday, 10/29/12, 11:59pm

In this assignment, you will design 3 computer programs using sequential, conditional, and looping logic. For each problem, you will draw the flowchart for each solution, as well as specify the solution in plain English/pseudocode. Some people think visually and have an easier time drawing the flowchart first, and then, they put words to the picture. Others think verbally, and therefore, it is easier to express a solution in words. Here is a link for more information on different learning styles:

1. Design the computational solution for the following problem using a flowchart AND pseudocode. Suppose the video game machines at your local arcade output coupons depending on how well you play the game. You can redeem 10 coupons for a candy bar or 3 coupons for a gumball. You prefer candy bars to gumballs.

   Design a solution that takes the amount of coupons won as input and outputs how many candy bars and gumballs you can get, when you spend all of your coupons on candy bars first and any remaining coupons on gumballs.

2. Design the computational solution for determining whether an object is buoyant using a flowchart AND pseudocode. Buoyancy is the ability of an object to float. Archimedes’ Principle states that the buoyant force is equal to the weight of the fluid that is displaced by the submerged object. The buoyant force can be computed by

   \[ F_b = V \times \gamma \]

   where \( F_b \) is the buoyant force, \( V \) is the volume of the submerged object, and \( \gamma \) is the specific weight of the fluid. If \( F_b \) is greater than or equal to the weight of the object, then it will float, otherwise it will sink.

   Design a solution that takes the weight (in pounds) and radius (in feet) of a sphere as input and outputs whether the sphere will sink or float in the water. Use \( \gamma = 62.4 \text{ lb/ft}^3 \) as the specific weight of water, and the volume of the sphere is \( \frac{4}{3} \pi r^3 \).

3. Design the computational solution for the Babylonian algorithm using a flowchart AND pseudocode. The Babylonian algorithm computes the square root of a positive number, \( n \), as follows:
   1. Make a guess at the answer (you can pick \( n/2 \) as your initial guess).
   2. Compute \( r = n / \text{guess} \)
   3. Set \( \text{guess} = (\text{guess} + r) / 2 \)
   4. Go back to step 2 for as many iterations as necessary. The more steps 2 and 3 are repeated, the closer guess will become to the square root of \( n \).
Design a solution that takes a positive number n as input and outputs the square root of n using the Babylonian algorithm.

Our next speaker is Dr. Don Morton, Arctic Region Supercomputing Center (ARSC), and he will talk about Weather Modeling. Since many of you have not heard of ARSC, do some research about the supercomputing center, as well as weather modeling, and add two questions for our next speaker to the end of your paper.

Electronically submit your document as a pdf by the assignment due date, using TEACH: https://secure.engr.oregonstate.edu:8000/teach.php?type=want_auth