Coding Assignment 6

For this assignment, you will be making two different types of Houses – one that gives out candy and one that gives out toothbrushes. Please watch the video of the new version of our Trick or Treating program before continuing with reading the directions for this assignment.

When a trick or treater goes to a Candy House, a '+' sign shows for the doorbell. When a trick or treater goes to a Toothbrush House, a '-' sign shows for the doorbell. A Candy House gives the trick or treater candy and the Toothbrush House gives the trick or treater a toothbrush. When the trick or treaters show the contents of their buckets, the number of toothbrushes should match the number of Toothbrush houses they visited.

Code6.cpp - Randomly decide if a house is a candy house or a toothbrush house. Because of polymorphism, both candy houses and toothbrush houses can both be stored in the vector of House pointers. This is the ONLY change to Code6.cpp.

TrickOrTreater.h/TrickOrTreater.cpp – No changes.

House.h – make House an abstract class.

House.cpp – remove the body of ringDoorbell().

Add two new derived classes of House – CandyHouse and ToothbrushHouse.

CandyHouse – Create a .h and .cpp file. Add CandyHouse to your makefile. CandyHouse's version of `ringDoorbell()` will add a '+' to the trick or treater's path. CandyHouse will pass back candy.

ToothbrushHouse – Create a .h and .cpp file. Add ToothbrushHouse to your makefile. Toothbrush's version of `ringDoorbell()` will add a '-' to the trick or treater's path and will pass back a TOOTHBRUSH instead of candy.

Create explicit destructors for House, CandyHouse and ToothbrushHouse. The destructors should print to the screen as shown in the output.

When the trick or treaters display their buckets, both candy and toothbrushes should be displayed. The number of toothbrushes in each bucket should match the number of houses that gave out toothbrushes.