Exam 2
ECE330, Spring 2014

In this exercise, you will prepare two projects that do identical work, using identical object relationships. One will be in C++, the other in Ruby. See the next page for a a screen shot of the directory structure you should end up with. The C++ Makefile is included to get you started. Use it! It has an option set that will make your life easier. Each project will implement a "main" program that shows off the behavior of the classes.

There will be three classes; all their data members are private and their function members (methods) are public:

- Shape (base class):
  - Member data: *color*, as a string
  - Member data: *border*, as a floating-point number
  - Member function (method): *draw*
    - When *draw* is called on an object of the base class shape, it prints "No information to draw myself."
  - Accessor functions *getColor, getBorder*
  - A constructor that sets its *color* and *border*.
- Circle (derived from Shape):
  - Member data: *radius*, as a floating-point number
  - Overridden function *draw*
    - When *draw* is called on an object of the class Circle that has radius 3.0, *color* green, and border 1.5, it prints "green Circle of radius 3.0 and border 1.5."
  - Accessor function *getRadius*
  - A constructor that sets its *color*, *border*, and *radius*.
- Square (derived from Shape):
  - Member data: *side*, as a floating-point number
  - Overridden function *draw*
    - When *draw* is called on an object of the class Square that has *side* 3.0, *color* green, and *border* 1.5, it prints "green Square of side 3.0 and border 1.5."
  - Accessor functions *getRadius*
  - A constructor that sets its *color*, *border*, and *side*.

The main program should declare two Circles, two Squares, and one Shape, with data as follows, and then call their *draw* functions in this order:

Circle: green, 1.5, 3.0
Circle: red, 2.0, 5.0
Square: blue, 1.0, 4.0
Square: purple, 2.5, 4.5
Shape: black, 1.0

N.B.: The number of decimal points for floating point numbers is not important.

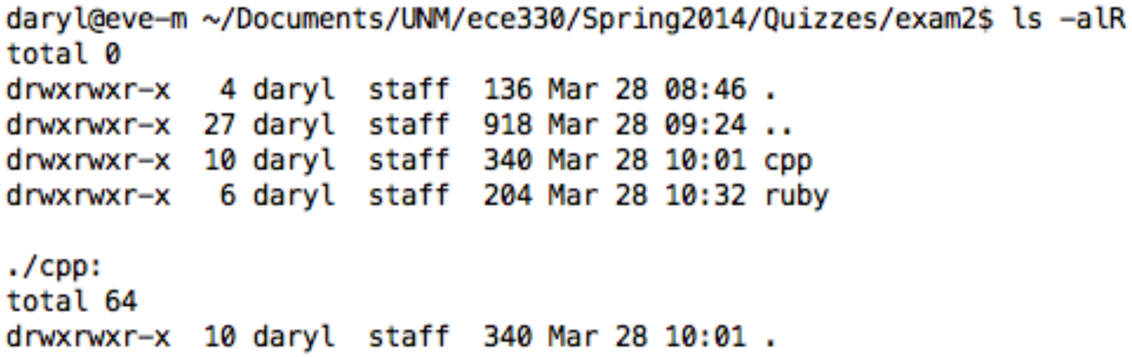
N.B.: Remember that versions of Ruby earlier than 1.9 (such as the university Linux server's) use the `require` keyword, later ones use the `require_relative` keyword.

Upon completion, clean your folder of "build artefacts", which are object files and executables (C++ folder only), and editor backups (if you are using Vim or Emacs these are the files with a ~ on the end of the filename). Then zip the entire folder at the top with this command:

$ zip –r exam2_${USER}.zip exam2

and post it to BBLearn.

Directory structure:



```
daryl@eve-m ~/Documents/UNM/ece330/Spring2014/Quizzes/exam2$ ls -alR
total 0
drwxrwxr-x   4 daryl  staff  136 Mar 28 08:46 .
drwxrwxr-x  27 daryl  staff  918 Mar 28 09:24 ..
drwxrwxr-x  10 daryl  staff  340 Mar 28 10:01 cpp
drwxrwxr-x   6 daryl  staff  204 Mar 28 10:32 ruby

./cpp:
total 64
drwxrwxr-x  10 daryl  staff  340 Mar 28 10:01 .
drwxrwxr-x   4 daryl  staff  136 Mar 28 08:46 ..
-rw-rw-r--   1 daryl  staff  494 Mar 28 09:59 Makefile
-rw-rw-r--   1 daryl  staff  274 Mar 28 10:00 circle.cpp
-rw-rw-r--   1 daryl  staff  315 Mar 28 09:54 circle.h
-rw-rw-r--   1 daryl  staff  500 Mar 28 09:53 main.cpp
-rw-rw-r--   1 daryl  staff  239 Mar 28 09:58 shape.cpp
-rw-rw-r--   1 daryl  staff  406 Mar 28 09:41 shape.h
-rw-rw-r--   1 daryl  staff  268 Mar 28 10:00 square.cpp
-rw-rw-r--   1 daryl  staff  308 Mar 28 09:58 square.h

./ruby:
total 32
drwxrwxr-x  6 daryl  staff  204 Mar 28 10:32 .
drwxrwxr-x  4 daryl  staff  136 Mar 28 08:46 ..
-rw-rw-r--  1 daryl  staff  300 Mar 28 10:31 circle.rb
-rw-rw-r--  1 daryl  staff  338 Mar 28 10:20 main.rb
-rw-rw-r--  1 daryl  staff  238 Mar 28 10:13 shape.rb
-rw-rw-r--  1 daryl  staff  288 Mar 28 10:30 square.rb
daryl@eve-m ~/Documents/UNM/ece330/Spring2014/Quizzes/exam2$
```