CS16, 10S, **H17**, due **Fri Lab 05.07**—Even More on Structs (handout)—Total Points: 50

Available online as http://www.cs.ucsb.edu/~pconrad/cs16/10S/homework/H17—printable PDF

Name: (4 pts) _____ Umail Address: (4 pts) _____ @umail.ucsb.edu

Lab Section (2 pts)—circle one: 9am 10am 11am noon unknown

(Note: For now, circle the lab section you are registered for on GOLD. If you need to request attendance at a different lab section because of an ACTUAL SCHEDULE CONFLICT, please email pconrad@cs.ucsb.edu with details)

This assignment is due **IN Lab on Friday, 05.07.**
It may ONLY be submitted Lab, in ESB1003 (Cooper Lab) at 9am, 10am, 11am or noon on Friday.
You must come IN PERSON to turn it in during your assigned Lab section.

Late Policy: No email submission allowed—and don't "slip it under my door". If you need to make it up, you must do so during office hours, or make an appointment to see me, and you must request this appointment within 48 hours of when the assignment was originally due.

Personal Day/Sick Day policy: Everyone is permitted one "personal day/sick day" when you get to make up a missed homework assignment for free during office hours or via appointment. After that, you may not make up the homework assignment—you can only earn back the points through extra credit opportunities.

(For more details, see the syllabus and the homework policy)

For this homework, the preparation is material on the following handout:
http://www.cs.ucsb.edu/~pconrad/cs16/10S/homework/H17/handout (pdf link)

Be sure to read not only the main text on the handout,
but also the little boxes off to the side, like the one see to the right of this sentence.

Once you've read that handout, write answers to the questions on this sheet (use the PDF link to print a copy of this if you weren't in class).

The little boxes on the handout—boxes like this one—also have important information you may need to complete the assignment.

```
struct Point
{
  double x;
  double y;
};
```

1. Using the struct Point declaration shown at the right of this page:
   a. (10 pts) Declare an array of struct Point that can hold 5 points.

   b. (10 pts) Write a for loop that assigns the x values of each of the 5 points to the following numbers:

      ```
      0.0, 1.0, 2.0, 3.0, 4.0
      ```

      and that assigns the y value of each point to the x value squared.

      Note that the index of an array should be an integer—so use an int variable to index into the array—but that you can assign an int value to a double variable, and it will be automatically converted into a double.

Please turn over for more...

...continued from other side

2. (5 pts) Write a struct definition that is an abstraction of a complex number.

   As a reminder: a complex number is a number of the form $a + bi$, such as $3 + 5i$.

   It has two parts:
   - a real part, represented by a
   - an imaginary part represented by b, which is a real coefficient of i, the square root of -1.

   Your struct definition should be capable of representing complex numbers such as $2.5 + 0.5i$

3. (5 pts) Write a struct definition that is an abstraction of a quadratic equation of the form $f(x) = ax^2 + bx + c$.

   Your abstraction should be able to store the values of a, b and c together in one struct—each of which can be any real number.

4. (10 pts) In the quadratic formula, if the discriminant is negative, then the quadratic equation will have two complex roots, with values as follows:

$$\frac{-b}{2a} + i\frac{\sqrt{4ac - b^2}}{2a} \quad \text{and} \quad \frac{-b}{2a} - i\frac{\sqrt{4ac - b^2}}{2a},$$

   Formula images from Wikipedia page for Quadractic Equation, used under CC-BY-SA license terms.

   where the $-b/2a$ part is the real part, and the other part is the coefficient of the imaginary part.

   Write the definition of a function called firstComplexRoot

   - takes, as its parameter, a variable that represents a quadratic equation (use your abstraction from question 3), and
   - returns a value that represents the first of these two roots (the one on the left, with the + sign between the terms), as a complex number (use your abstraction from question 2).

   Keep in mind that both of your struct definitions may have members called a and b—so it will be important to keep track of any use of the letter a refers to a value inside the quadratic equation struct that you passed in to the function, or the a inside the complex value that is being returned.

End of H17