CS 115 Exam 3, Spring 2010

Your name: _____

Rules

- You must briefly explain your answers to receive partial credit.
- When a snippet of code is given to you, you can assume
  - that the code is enclosed within some function, even if no function definition is shown
  - that the `main` function is properly defined
  - that the `iostream`, `fstream`, `string`, `cstring`, and `cmath` libraries have been included at the beginning of the program.
- When you are asked to write *a snippet* of code, you may assume
  - that your code is enclosed within some function
  - that any necessary libraries have been included.
- When you are asked to write *a complete program*, you must write the `#include` statements, the `int main()`, etc. in your solution to receive full credit.
- A line consisting solely of "..." represents one or more unspecified C++ statements, some of which may change the values of program variables.

Grade (instructor use only)

	Your Score	Max
Problem 1		15
Problem 2		15
Problem 3		10
Problem 4		15
Problem 5		30
Problem 6		15
Total		

Problem 1: 15 points.

Parts (a)-(d): Match the following descriptions with the terms they describe by writing those terms in the space provided. The choices of terms are:

- this
- constructor
- destructor
- class
- struct
- object
- method
- pointer
- overloading
- operator
- bubble sort
- selection sort
- linear search
- binary search
- reference
- dereference

Not all terms will be used.

(a) An algorithm that puts an array in order by finding the smallest (or largest) element and getting it into place, then finding the second-smallest element, then the third-smallest, etc.

(b) A function that is called automatically when an object is declared

(c) A variable containing the address of another variable

(d) A user-defined data type whose members are <u>all public</u> and can all be accessed directly with the dot-notation

Part (e):

Given the following code:

```
int num;
int* p = &num;
…
```

Write code that will do the same thing as

```
cout << num << endl;
```

without directly using the variable name `num`.

Problem 2: 15 points.

Answer the following questions about the binary search code on the next page, which works identically to the binary search code from your labs.

ARRAY CONTENTS

Chuao	Dagoba	Godiva	Hershey	Lindt	Tcho	Valrhona	Vosges

Fill out the following table tracing a binary search for *Scharffen-Berger* in this array. You should fill out one row per iteration of the loop. If there are more rows than iterations, leave the extra rows blank.

In the *Compare To:* column, you should give the VALUE (Chuao, Dagoba, etc.) that will be compared to Scharffen-Berger.

Old value of `first`	**Old value of `last`**	**Compare to:**	**New value of `first`**	**New value of `last`**
0	7			

Code for Problem 2.

You may tear this page out of your exam.

In this code:

- `arr` is our 8-element array
- `size` is 8
- `searchStr` is "Scharffen-Berger"

```
int first = 0;
int last = size - 1;
int middle;
bool found = false;

while ( first <= last and found == false) {

        middle = (first + last) / 2;

       if ( searchStr == arr[middle] ) {
              found = true;
       }
       else if ( searchStr < arr[middle] ) {
              last = middle - 1;
       }
       else {
              first = middle + 1;
       }

}
```

Problem 3: 15 points.

Assume the following:

- A C++ class called `Person` has been defined for you
- The `cstring` class has been included
- `Person` has 2 private data members:
  - `char firstname[25];`
  - `char lastname[25];`

a) Finish defining the copy constructor for `Person`:

```
// Copy the first name and last name of the other Person
Person::Person( const Person& other) {









}
```

b) If `p1` and `p2` are two variables of type `Person`, we would like `(p1 > p2)` to be `true` if `p1`'s last name comes after `p2`'s last name alphabetically OR if their last names are the same and `p1`'s first name comes after `p2`'s first name.

```
/* If our last name comes after the other last name,
 * or if our last names are tied and our first name
 * comes after the other first name, return true.
 * Otherwise, return false. */
bool Person::operator > ( const Person& other) const {









}
```

Problem 4: 15 points.

For this problem, you must write a **class definition** for a class named **`Circle`** that contains the elements listed below.

Note that data members should be private and member functions should be public.

You do not need to write the definitions of the member functions – just the prototypes. You will define the functions in the next problem.

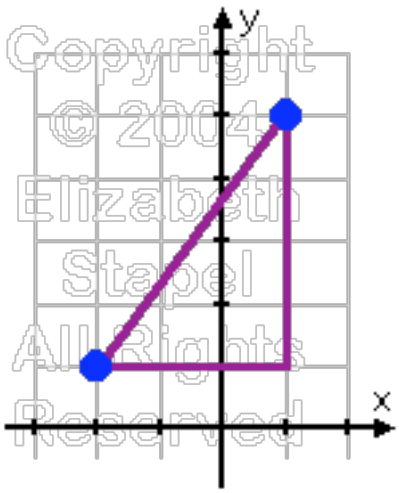
- The *x* and *y* coordinates of the center of the circle (both as `float`)
- The radius of the circle (as a `float`)

- Prototype for a default constructor
- Prototype for a function called `SetCenter`. This function will take two numeric values as inputs. It will not return anything.
- Prototype for a function called `SetRadius`. This function will take a numeric value as an input and return a `bool`.
- Prototype for a function called `GetArea` that will return the area of the circle
- Prototype for a function called `IsInsideCircle`. This function will take two numbers x1 and y1 as parameters and return `true` if the point (x1, y1) is inside the circle.

Problem 5: 30 points.

In this problem, you will write definitions for the functions in the class `Circle`. Here is a little bit more information about the functions. You may assume that `cmath` has been included. *Note that none of your code for this problem should include* `cin` *or* `cout` *statements!*

- The default constructor will initialize the center coordinates and the radius to 0.
- The `SetCenter` function will set x equal to its first input and y equal to its second input.
- The `SetRadius` function will:
  - return `false` if its input is less than 0.
  - otherwise, set the radius equal to its input parameter and return true.
- The `GetArea` function will return the area of the circle. Remember that the area is equal to pi times the square of the radius.
- The `IsInsideCircle` function should return `true` if the input point (x1, y1) is inside the circle and `false` otherwise. Here's how to figure that out:
  - If the distance between (x1, y1) and the center of the circle (x, y) is smaller than the radius, then the point is inside the circle.
  - To get the distance between (x1, y1) and (x, y), you can imagine a right triangle whose hypotenuse is the distance.



Copyright © 2004 Elizabeth Stapel All Rights Reserved

In other words:

$(x-x1)^2 + (y-y1)^2 = distance^2$

You can take it from here ☺

Problem 6: 15 points.

Assume that the class definitions you wrote in Problems 4 and 5 are located in a file called `Circle.h` in the same directory as the program you're about to write.

Write **a complete program** that uses the `Circle` class from `Circle.h` to do the following:

- Create a `Circle` object. Ask the user to supply the center coordinates and the radius. Print an error message and exit the program if the radius is negative.
- Set the center and radius of your object.
- Use the `GetArea` function to calculate the area of this circle, and print it
- Ask the user to supply the x and y coordinates of another point. Create a new circle with the same radius and with the center at these new coordinates.
- If the center of one circle is inside the other circle, print a message:
  Your points are too close together!

REFERENCE

C-string functions:

`strlen`	Input is a C-string. Returns the length of the string (not including the null terminator). Usage example: `length = strlen(name);`
`strcat`	Input is two C-strings. Appends the second string to the end of the first string (the first string is changed, but the second is not). Usage example: `strcat(str1, str2);`
`strcpy`	Input is two C-strings. Copies the second string to the first string, overwriting the original contents. Usage example: `strcpy(str1, str2);`
`strcmp`	Input is two C-strings. Returns 0 if they are the same, a negative number if `str2` is alphabetically greater than `str1`, and a positive number if `str1` is greater than `str2`. Usage example: `if(strcmp(str1, str2) > 0)`