Homework 8: Classes

CS16 - Summer 2021

Due:	Thursday, August 19, 2021 (11:59 PM PDT)
Points:	70
Name:	_____
Homework buddy:	_____

- You may collaborate on this homework with **at most** one person, an optional "homework buddy."
- **Submission instructions:** All questions are to be written (either by hand or typed) *in the provided spaces* and turned in as a single PDF on Gradescope. In other words, you must edit this file directly! Reach out on Slack if you want some suggestions on how to do this. Do not copy and paste the text into a word processor; we will not accept this and your homework may not be graded. If you submit handwritten solutions, write legibly. We reserve the right to give 0 points to answers we cannot read.

1. (2 points) What's the main difference between a `struct` and `class` in C++?

2. (5 points) What's the difference between `public` and `private` members of a class in C++?

3. (3 points) What are class constructors?

4. (24 points) Suppose your program contains the following class definition:

```
class Point {
   public:
      Point(double n1, double n2);
      Point(); // initializes member variables to 0
      double get_x(); // returns value of x
      double get_y(); // returns value of y
      void set_x(double n); // sets a new value for x
      void set_y(double n); // sets a new value for y
   private:
      double x, y;
};
```

a. (12 points) Given the comments shown, give definitions to all 6 of these member functions/constructors:

For points $(x_1, y_1), (x_2, y_2)$, the Euclidean distance formula is given by:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Suppose we want to add a member function to the `Point` class that computes the distance between a given point and itself. Call it `distanceFrom`. The function should take as argument *another* object of type `Point` and return the computed distance. Assume that the `<cmath>` library is already included.

b. (2 points) Give the member function *declaration* for the `distanceFrom` member function.

c. (4 points) Give the member function *definition* for `distanceFrom`.

For a point (x, y), we can rotate it by θ degrees to obtain a new point (x', y'):

$$x' = x\,cos(\theta) - y\,sin(\theta)$$
$$y' = x\,sin(\theta) + y\,cos(\theta)$$

Suppose we want to add a member function to the `Point` class that rotates the point by a given degree and *updates* the values for the member variables `x` and `y`. Call it `rotate`. The function should take as argument a double representing the degree θ. Assume that the `<cmath>` library is already included.

d. (2 points) Give the member function *declaration* for the `rotate` member function.

e. (4 points) Give the member function *definition* for `rotate`.

5. (2 points) What are derived classes and what mechanism do they use in order to fulfill what they need to do?

6. (2 points) Can a derived class directly access by name a private member variable of the parent class?

7. (2 points) Suppose the class `SportsCar` is a publicly derived class of a class `Automobile`. Suppose also that the class `Automobile` has public member functions named `accelerate` and `addGas`. Will an object of the class `SportsCar` have member functions named `accelerate` and `addGas`?

8. (14 points) Suppose your program contains the following class definition:

```
class Automobile {
   public:
      void set_price(double new_price);
      void set_profit(double new_profit);
      double get_price();
   private:
      double price;
      double profit;
      double get_profit();
};
```

Suppose the main part of your program contains the following declaration and that the program somehow sets the values of all the member variables to some values:

```
Automobile hyundai, jaguar;
```

Which of the following statements are then **not** allowed in the main part of your program and explain **why**.

(a) `hyundai.price = 4999.99;`
(b) `jaguar.set_price(30000.97);`
(c) `double a_price, a_profit;`
(d) `a_price = jaguar.get_price();`
(e) `a_profit = jaguar.get_profit();`
(f) `a_profit = hyundai.get_profit();`
(g) `if (hyundai == jaguar) {hyundai = jaguar;}`

9. (16 points) Suppose your program contains the following class definition:

```
class TwoNumbers {
   public:
      TwoNumbers(int n1, int n2);
      TwoNumbers(); // initializes num1, num2 to 0
      double sum(); // returns sum of num1 & num2
      double difference(); // returns diff. of num1 from num2
      double max(); // returns larger of num1, num2
   private:
      double num1, num2;
};
```

a. (10 points) Given the comments shown, give definitions to all 5 of these member functions/constructors:

b. (2 points) Consider these instructions in `main()`:

```
TwoNumbers thisOne, thatOne(5,7);
thisOne.num1++;
thisOne.num2 -= 7;
thatOne.num2 = thatOne.sum() + thisOne.difference();
cout << thisOne.max() / thatOne.max();
```

Explain all the reasons **why** this code will not compile.

c. (2 points) What would you change to the class definition to make this code compile?

d. (2 points) When you fix it, what would these instructions do?