CMPSC 24 Midterm-I
Spring 2017
4/26/2017

This exam is closed book, closed notes. You may use a one-page A4 size paper containing your nodes. Write your name on the your notes paper and all other sheets of the exam. No calculators or phones are allowed in the exam. **Write all your answers on the answer sheet. All exam sheets, notes paper and scratch paper must be turned in at the end of the exam.**

By signing your name below, you are asserting that all work on this exam is yours alone, and that you will not provide any information to anyone else taking the exam. In addition, you are agreeing that you will not discuss any part of this exam with anyone who is not currently taking the exam in this room until after the exam has been returned to you. This includes posting any information about this exam on Piazza or any other social media. Discussing any aspect of this exam with anyone outside of this room constitutes a violation of the academic integrity agreement for CMPSC 24.

Signature: ____

Name (please print clearly):____

Umail address: _____________________________@umail.ucsb.edu

Perm number: ____

Your exam hall: ____

Name of student to your left ____

Name of student to your right _______________________________________

You have 75 minutes to complete this exam. Work to maximize points. A hint for allocating your time: if a question is worth 10 points, spend no more than 5 minutes on it if a question is worth 20 points, spend no more than 10 minutes on it etc. If you don’t know the answer to a problem, move on and come back later. Most importantly, stay calm. You can do this.

WRITE ALL YOUR ANSWERS IN THE PROVIDED ANSWER SHEET IN PEN!

DO NOT OPEN THIS EXAM UNTIL YOU ARE INSTRUCTED TO DO SO.
GOOD LUCK!

Name: ______________________

Part 1 [10 points] Multiple Choice: ***Select the SINGLE best answer by filling in the circles provided in your answer sheet. You must shade your selection completely as shown below:***

○ Not selected ● Selected

1. Which of the following information must be known to the *user* of an abstract data type (ADT) for using the ADT correctly?
   A. The public methods including the implementation details of each method
   B. The public methods, including the pre and post conditions for each method but not the implementation details
   C. The private and public methods and member variables of the class, and the implementation of all methods.
   D. None of the above

2. Describe a technique for incorporating the principle of information hiding when describing a function's behaviour.
   A. Value semantics
   B. Private and public access-control specifiers
   C. Pre and post conditions
   D. None of the above

3. How does the compiler handle classes that do not define any explicit constructor?
   A. The compiler produces a compile time error because every class must have a constructor explicitly defined.
   B. The compiler will automatically generate a default constructor (that takes no parameters) and a default copy constructor (that takes another object of the class as parameter)
   C. The compiler will automatically generate an appropriate constructor (parameterized or non-parameterized), depending on how objects are created by any code that uses the class.
   D. None of the above

4. Friend functions of a class are typically written only by the programmer who implements a class because:
   A. Friend functions are private methods of a class and can only be accessed by the programmer who implements the class
   B. Friend functions can access the private member variables of the class.
   C. Friend functions are the only way the programmer of a class can overload operators for that class
   D. None of the above

5. Consider the following declaration of the non-member function: swap for the point class. Assume that a correct definition of the swap function and the point class exists elsewhere:

```
void swap (point &p1, point &p2);
```

Which of the following C++ constructs is used to pass parameters to the swap function.

A. Default assignment operator of the point class
B. Default copy constructor of the point class
C. Default constructor of the point class
D. None of the above

Part 2 [40 points] Consider the following definition of the point class.

```
class point{
   public:
        double get_x() const {return x};
        double get_y() const {return y};
        void shift(const point &del_p);
        //Postcondition: The point has been moved in the x-axis and
        //y-axis by del_p's x and y coordinates respectively

   private:
        double x; // x-coordinate of the point
        double y; // y-coordinate of the point
};
// Non-member functions
void swap (point &p1, point &p2);
// Post condition: Swaps the x and y coordinates of p1 with the x
// and y coordinates of p2
```

1. **[2 pts]** Which of the following is true about the definition of the point class above? *Select the single best answer by shading it in your answer sheet*
   A. An object of the point class can be created as follows: point p(10, 20);
   B. When a point object is created using the default constructor, the x and y member variables are initialized to 0
   C. The `shift()`method cannot modify the x and y member variables of its argument (del_p)
   D. None of the above

2. **[2 pts]** Which of the following methods of the point class CAN modify the x and y member variables **of the object that activates the method**? *Select the SINGLE best answer by shading the appropriate circle in your answer sheet.*
   A. `get_x()`
   B. `shift()`
   C. Both A and B
   D. None of the above

You have been provided with some code below that uses the point class (from the previous page). You have also been given the expected output of the code. With the current definition of the point class, the code will not produce the expected output. You may assume that the code is embedded in an otherwise correct and complete C++ program. Based on the given information answer questions 3-6

```
# include <iostream>
#include “point.h”
using namespace std;

void printPoint(const point& p1, const point& p2){
      cout<<“ P1:”<< p1.get_x()<<" "<<p1.get_y()<<endl;
       cout<<“ P2:”<< p2.get_x()<<" "<<p2.get_y()<<endl;
 }

int main(){
      point p1(100.0), p2(50.0, 60.0);
      cout<< “Initial value of points”<<endl;
      printPoint(p1, p2);

      cout<< “\nShift P1 by the amount specified in P2”<<endl;
      p1.shift(p2);
      printPoint(p1, p2);

      cout<< “\nSet P1 to double the value of P2”<<endl;
      p1 = 2* p2;
      printPoint(p1, p2);

      cout<< “\nSwap P1 and P2”<<endl;
      swap(p1, p2);
      printPoint(p1, p2);

      cout<< “\nSet P1 equal to P2”<<endl;
      p1 = p2;
      printPoint(p1, p2);
}
```

Expected output:

```
Initial value of points:
P1:100.0 0.0
P2: 50.0 60.0

Shift P1 by P2
P1:150.0 60.0
P2: 50.0 60.0

Set P1 to double the value of P2
P1:100.0 120.0
 P2: 50.0 60.0

Swap P1 and P2
 P1: 50.0 60.0
P2:100.0 120.0

Set P1 equal to P2
P1:100.0 120.0
P2:100.0 120.0
```

3. **[6 pts]** Rewrite the DEFINITION of the point class so that the code in the previous question compiles correctly and produces the expected output. Do not provide the implementation of any of the methods (other than the constructor) in this part of the question. Just write the class definition
    - DO NOT delete or modify any of the provided methods
    - Add the following elements to the point class to complete the definition:
        - A single constructor implemented in line with default parameters (0.0, 0.0)
        - All operators that need to be overloaded, in particular the * operator. You must decide whether you need to make it a member function, a non-member or a friend function based on the usage of the point class code on the previous page

4. **[10 pts]** Implement the `shift()` member function in your answer sheet.
5. **[10 pts]** Implement the `swap()` non-member function in your answer sheet.
6. **[10 pts]** Implement the overloaded * operator for the point class to match your definition from Q3

Scratch Paper