**Homework 1**

For problems 1 through 4, explain why the code as shown is almost certainly not what the programmer intended, and how it should be fixed to work the way the programmer probably had in mind.

1. (10 pts) What is wrong with the following program and how should it be fixed?

1  public class MyClassA {  
2    int v = 12;  
3   
4    public MyClassA (int pV) {  
5      v = pV;  
6    }   
7   
8    public static void main (String args []) {  
9      MyClassA m = new MyClassA ();  
10   } // end main  
11 } // end class MyClassA

**In line 9 When creating an object for MyClassA needs to a single int parameter to match the constructor. The programmer can add a default constructor or delete the one with the int parameter constructor.**

2. (10 pts) What is wrong with the following program and how should it be fixed?

1  public class MyClassB {  
2    int v = 12;  
3   
4    public void MyClassB (int pV) {  
5      v = pV;  
6    }   
7   
8    public static void main (String args []) {  
9      MyClassB m = new MyClassB (23);  
10   } // end main  
11 } // end class MyClass

**MyClassB only has a default constructor with no parameter. To fix this the programmer can delete the void on line 4 and turn it from a method to a constructor that accepts an int.**

3. (10 pts) What is wrong with the following program and how should it be fixed?

1   public class MyClassD {  
2     public static void main (String args []) {  
3       MyClassC m = new MyClassC (23);  
4     } // end main  
5   } // end class MyClassD  
6   
7   class MyClassC {  
8     int v = 12;  
9   
10    public MyClassC (int pV) {  
11      int v = pV;  
12    }   
13   
14  } // end class MyClassC

**While this code would run, it’s pretty cluttered since all MyClassD does is create an instance of MyClassC. Simply taking out the middle man and putting the code from MyClassD into MyClassC would compress and streamline the code.**

4. (10 pts) What is wrong with the following program and how should it be fixed?

1   public class MyClassE {  
2     public static void main (String args []) {  
3       MyClassF m = new MyClassF (23);  
4     } // end main  
5   } // end class MyClassE  
6   
7   class MyClassF {  
8     int v = 12;  
9   
10    private MyClassF (int pV) {  
11      v = pV;  
12    }   
13   
14  } // end class MyClassF

**MyClassF has a private constructor and can’t accessed from MyClassE. To fix this the programmer can just change the MyClassF constructor to public .**

5. (10 pts) Given all the problems identified in problems 1 through 4, explain in detail why the following code works, ie, compiles without errors or warnings.

1  public class MyClassG {  
2    public static void main (String args []) {  
3      MyClassH m = new MyClassH (23, true);  
4    } // end main  
5  } // end class MyClassG  
6   
7  class MyClassH {  
8    int v = 12;  
9   
10   public MyClassH (int x, boolean b) {  
11     this (x);  
12   }   
13   
14   private MyClassH (int pV) {  
15     v = pV;  
16   }   
17   
18 } // end class MyClassH

**Based on the previous questions, it’s safe to say the code works because the MyClassH is visible to MyClassG, even though there’s a private constructor as well. The specific constructor being called with correct types of parameters is the public not the private.**

6. (10 pts) Explain why the following class hierarchy is not reasonable:

* DefenseDepartment
  + General
    - Private

**There are too many different variables within each class that would not be compatible with the parent nor child classes. A Defense department wouldn’t have many things applicable to a General class, and although a Private class may have similar things to the General class, there still isn’t much that can be taken from the parent class since there isn’t much in common besides maybe a name variable, among other things.**

7. (10 pts) Give at least one example of a reasonable field for each of the following classes in the following class hierarchy. Be sure that the field is at the right level in the hierarchy.

* Vehicle – **public String ownerName**
  + Car – **public int mileage**
  + Airplane – **public String name**
    - Passenger – **public int seatNum, public boolean firstClass**
    - Fighter – **public double topSpeed**
    - Bomber – **public int range**
  + SpaceShip – **public String type**

8. (10 pts) Give at least one example of a reasonable method for each of the following classes in the following class hierarchy. Be sure that the method  is at the right level in the hierarchy. Constructors, getters and setters don't count for this problem.

* Vehicle – **public String toString()**
  + Car – **public int addMileage(int miles)**
  + Airplane– **public void removeSeatAvailability(Passenger p)**
    - Passenger– **public int frequentFlyer(int f)**
    - Fighter– **public void shootGun(int bullets)**
    - Bomber – **public void useBomb(int bomb)**
  + SpaceShip– **public boolean isAirborne()**

9. (10 pts) Are a Private and a Platoon in an encapsulation or an inheritance relationship? Explain

**Private and Platoon are in an encapsulation because they are both sub units that could be added to an Army class. Its main uses would be to enable polymorphism and to be able to reuse code for different classes by putting it in a common superclass.**  
  
10. (10 pts) Present reasonable parent and child classes for the class Tree (the biological kind). Give a short explanation for why the classes you are proposing are in good parent-child relationships.

* **Tree**
  + **Branches**
    - **Leaves**

**The child classes all have properties inherited from their parent classes. Branches inherits variables from Tree, and Leaves from Branches.**

**Grading Rubric:**

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Meets** | **Does not meet** |
| Problem 1 | **10 points** Explains why the code as shown is almost certainly not what the programmer intended.  Explains how it should be fixed to work the way the programmer probably had in mind. | **0 points** Does not explain why the code as shown is almost certainly not what the programmer intended.  Does not explain how it should be fixed to work the way the programmer probably had in mind. |
| Problem 2 | **10 points** Explains why the code as shown is almost certainly not what the programmer intended.  Explains how it should be fixed to work the way the programmer probably had in mind. | **0 points** Does not explain why the code as shown is almost certainly not what the programmer intended.  Does not explain how it should be fixed to work the way the programmer probably had in mind. |
| Problem 3 | **10 points** Explains why the code as shown is almost certainly not what the programmer intended.  Explains how it should be fixed to work the way the programmer probably had in mind. | **0 points** Does not explain why the code as shown is almost certainly not what the programmer intended.  Does not explain how it should be fixed to work the way the programmer probably had in mind. |
| Problem 4 | **10 points** Explains why the code as shown is almost certainly not what the programmer intended.  Explains how it should be fixed to work the way the programmer probably had in mind. | **0 points** Does not explain why the code as shown is almost certainly not what the programmer intended.  Does not explain how it should be fixed to work the way the programmer probably had in mind. |
| Problem 5 | **10 points** Given all the problems identified in problems 1 through 4, explains in detail why the code works, ie, compiles without errors or warnings. | **0 points** Given all the problems identified in problems 1 through 4, does not explain in detail why the code works, ie, compiles without errors or warnings. |
| Problem 6 | **10 points** Explains why the class hierarchy is not reasonable. | **0 points** Does not explain why the class hierarchy is not reasonable. |
| Problem 7 | **10 points** Gives at least one example of a reasonable field for each of the classes.  The field is at the right level in the hierarchy. | **0 points** Does not give at least one example of a reasonable field for each of the classes.  The field is not at the right level in the hierarchy. |
| Problem 8 | **10 points** Gives at least one example of a reasonable method for each of the classes.  The method is at the right level in the hierarchy.  Does not include constructors, getters and setters. | **0 points** Does not give at least one example of a reasonable method for each of the classes.  The method is not at the right level in the hierarchy.  Includes constructors, getters and setters. |
| Problem 9 | **10 points** Explains inheritance and encapsulation correctly and in sufficient detail given the example provided. | **0 points** Does not explain inheritance and encapsulation correctly and in sufficient detail given the example provided. |
| Problem 10 | **10 points** Presents reasonable parent and child classes for the class Tree.  Gives a short explanation for why the classes you are proposing are in good parent-child relationships. | **0 points** Does not present reasonable parent and child classes for the class Tree.  Does not give a short explanation for why the classes you are proposing are in good parent-child relationships. |