

CS211 Summer 2012

Dr. Kinga Dobolyi

Final Examination

Part 1: Short Answer.

1. Examine the code below, and answer the following questions. (53 points)

```
import java.io.*;
import java.util.ArrayList;

public abstract class MyClass{
    private static int age;
    public ArrayList salaries;

    public int compareTo(Object o){
        MyClass other = (MyClass)o;
        if (other.name == null)
            return 1;
        else
            return (Math.abs(name.compareTo(other.name)));
    }

    private boolean find(Integer wage){
        for(int i = 0; i < salaries.size(); i++)
            if (salaries.contains(wage))
                return true;
        return false;
    }

    private static void encode(int result){
        System.out.println(name);        //1
        System.out.println(age);         //2
        System.out.println(salaries);     //3
        System.out.println(result);      //4
    }

    protected void openFile(){
        BufferedReader buf = null;
        try {
            buf = new BufferedReader(new FileReader("input.txt"));
            System.out.println("great success!");
        }
        catch (Exception e) {
            System.out.println("bad file!");
        }
        finally {
            System.out.println("close buffer!");
        }
    }
}
```

CS211 Summer 2012

Dr. Kinga Dobolyi

Final Examination

Student Name: _____

Student G#: _____

Student signature for Honor Code:

- [illegible]

- e. Circle YES or NO whether the statements in the encode method (numbered 1 through 4) would cause a compilation error, and in ONE SENTNCE explain why (4 points)

Statement 1	YES	NO
-------------	-----	----

Statement 2	YES	NO
-------------	-----	----

Statement 3	YES	NO
-------------	-----	----

Statement 4	YES	NO
-------------	-----	----

- f. What will be the output of the **openFile** method when *input.txt* does not exist? (2 points)

- g. What will be the output of the **openFile** method when it is able to open the file *input.txt*? (2 points)

- h. If I wanted to change the **find** method to return the **Integer** found, rather than a boolean value, and I had the expression

salaries.get(i)

What **type** would this evaluate to at compile time? What **type** would this evaluate to at runtime? (2 points)

- [illegible]

2. Give the output for the following code (you can assume the **toString** method of **Object** returns a **String** such as "[0x0123456]" which refers to a memory address – we don't care what the actual value of the memory is):

```
import java.util.*;
public class Test{
    public static void main(String[] args){
        ArrayList list = new ArrayList();
        list.add(new Object());
        list.add(new String("Kinga"));
        list.add(new Integer(3));
        for(Object o : list)
            System.out.println(o.toString());
    }
}
```

- a. Write the output below: (3 points)

- b. Explain what polymorphism is in Java using the example above. (4 points)

3. Read all three parts below before starting to write your code. Your code must compile for full credit.
 - a. Write an interface called **Item**. An **Item** should have a void method called *print* to print out the item that uses the default **toString** methods to print out the fields of the class. (8 points)
 - b. Now write an abstract class called **Computer** that implements the interface from question 3-a. This class should store the brand of the computer as a **String**, as well as the price of a **Computer** as a primitive floating-point number. Any object that is a **Computer** must maintain the following invariants: the brand must be at least 3, and no more than 20 characters long, and the price must be a positive number at or below a thousand. Your code must implement these invariants for full credit. These fields should be mutable by other classes, however, the **Computer** class must exhibit the object-oriented properties of encapsulation and information hiding as we have discussed in class. (15 pts)

(continue answer to b here if needed)

- c. Finally, write a concrete class **MacBook** which extends the **Computer** class with an additional field called *applications* that stores a list of them (each application can be stored as a **String**). You must write a constructor that initializes all fields of the **MacBook** (including inherited ones), and you may not shadow any variables. Write a **toString** method to print out all of the fields of the **MacBook** (9 pts)

4. Circle TRUE or FALSE for the following statements, and defend your choice in ONE SENTENCE for full credit. (20 points)

a. The regular expression "a*b*c*" will match an empty String

TRUE FALSE

b. The final keyword can be applied to a class, and doing so means you cannot further extend the class

TRUE FALSE

c. Using generics in Java helps eliminate ClassCastExceptions

TRUE FALSE

d. You can run bytecode on an operating system that does not have a Java compiler installed

TRUE FALSE

e. An abstract class **must** have at least one abstract method

TRUE FALSE

f. Java uses static binding of methods, as it is a compiled language

TRUE FALSE

g. Overloading a method and overriding a method are mutually exclusive

TRUE FALSE