

CS125 Spring 2009 Exam 4 Review

Before you start, please note: it is possible to get strong clues for answering some of the questions in the review by referring to other questions. In general, you should not depend on both versions of a question showing up in the exam.

Also note: this study guide is not comprehensive. I strongly suggest reviewing: the practice problems posted online, homework assignments, material from the last exam & the self check questions in the text.

1. The hardware of a computer refer(s) to the physical parts of a computer system while software refer(s) to the intangible components.
2. When data is processed into a meaningful form, it becomes information.
3. The software we've programmed could be classified as application software since it performs specific tasks for the user.
4. The primary roles of operating system software are to run the computer & provide low level services to other programs.
5. There are 8 bits in a byte of memory.
6. ROM stands for Read Only Memory
7. RAM stands for Random Access Memory
8. How are the Web & the Internet related & how do they differ?

The Internet is the largest, most well-known computer network; the Web is the portion of the Internet that makes resources & information available through web pages.

9. Define the primary operations of a computer.

input – entering data into the computer
processing – performing operations on data
output – displaying results
storage – saving data for future use

10. List & briefly define the six categories into which computers are general classified.

Embedded computers	Mobile devices	Personal computers
Midrange servers	Mainframe computers	Supercomputers

11. List, in order, the steps of the machine processing cycle:

Fetch
Decode
Execute
Store

12. Compare & contrast the ALU with the FPU.

Both perform arithmetic operations (+ - × ÷) & comparisons (< ≤ ≥ > ≠) on data. However the FPU typically performs operations involving floating point (decimal) numbers (such as double & float in Java) while the ALU handles non floating point data (such as int & Boolean in Java)

13. Give an example of a URL & diagram its components.

http://www.cord.edu/faculty/pikalek/Pikalek-Vita-2008.pdf			
<u>protocol</u>	server	path	resource or filename

14. Given two classes, CheckingAccount & SavingsAccount which both inherit from a class BankAccount, place an --> by the statements that will print:

```
CheckingAccount ca = new CheckingAccount(100.0);
SavingsAccount sa = new SavingsAccount (200.0);
BankAccount ba = new BankAccount (300.0);
Object o1 = (Object)ca;
Object o2 = (BankAccount)o1;
```

```
--> if(ca instanceof CheckingAccount)
    System.out.println("ca instanceof CheckingAccount");

if(ca instanceof CheckingAccount)
    System.out.println("sa instanceof BankAccount");

--> if(ba instanceof BankAccount)
    System.out.println("ba instanceof BankAccount");

--> if(ba instanceof Object)
    System.out.println("ba instanceof Object");

--> if(o1 instanceof Object)
    System.out.println("o1 instanceof Object");

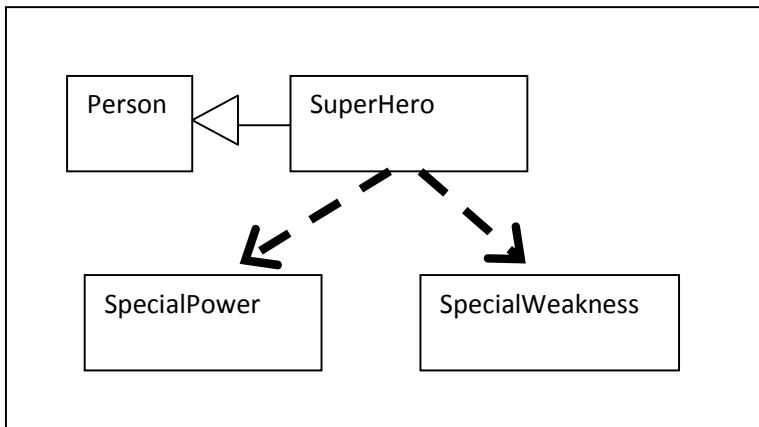
--> if(o2 instanceof CheckingAccount)
    System.out.println("o2 instanceof CheckingAccount");
```

15. Say we had the following class relationships (classes are in bold):

SuperHero is a type of **Person**
SuperHero has a **SpecialPower**
SuperHero has a **SpecialWeakness**

inheritance
instance field
instance field

Indicate which relationships denote inheritance & which denote instance fields. Draw out the corresponding UML diagram.



16. Consider the following class

```
public class RegularPerson
{
    private String name;
    public Parent(String n)
    {
        name = n;
    }
    public String getName()
    {
        return name;
    }
}
```

Make a new class called KnightedPerson that inherits from regular person. Assume the following bit of code will be used to test your class:

```
KnightedPerson k = new KnightedPerson("Jim", "Moorhead");
System.out.println( k.getName() );
System.out.println( "expected: Sir Jim of Moorhead" );
```

Make certain that you include documentation comments for your class & all of its methods/constructors.

```
/** models a Person who has been knighted
 *
 */
public class KnightedPerson extends Person
{
    String kingdom;

    /** Inits the KnightedPerson to have the given name & kingdom
     * @param n then name of the person
     * @param k the name of the kigndom
     */
    public KnightedPerson(String n, String k)
    {
        super(n);
        kingdom = k;
    }
    /**
     * Displays the info in the form of Sir <name> of <kingdom>
     */
    public void getName()
    {
        String result = "Sir " + super.getName() + " of " + kingdom;
        return result;
    }
}
```

17. What are the benefits of using inheritance in Java programming?

Benefits include: code reuse - since we don't have to rewrite code that already works

managing complexity - if we need to fix a bug in the parent class method, it's fixed in all the classes that inherit it

18. Write a bit of code that demonstrates how to get objects from two different types into the same array (hint: when we did this in class, the classes we used were Square & ImprovedSquare).

```
Object array = new Object[2];  
array[0] = new Square();  
array[1] = new ImprovedSquare();
```

19. Consider the following code. Add the code necessary so that the constructor for **ImprovedFoo** uses **a** to set its own instance field & uses **b** to set the instance field of its parent class **Foo**.

```
public class Foo  
{  
    private int x;  
    public Foo(int a)  
    {  
        x = a;  
    }  
}
```

```
public class ImprovedFoo  
{  
    private int y;  
    public Foo(int a, int b)  
    {  
        // your code goes right below here
```

```
        super(a);  
        y = b;
```

```
        // your code goes right above here
```

```
    }  
}
```