

Name _____ TA: Han [] Sherman []
Cooper [] Yang []

COMP114, Exam #1.

Thursday, February 7, 2001

Pledge: I have neither given nor received unauthorized aid on this exam.

(signed) _____

Please write your answers on the exam form. Closed book and notes. No calculator.

There are 9 pages on this exam.

Point values are in parentheses.

Write your name and check off your TA on each page (just in case the pages come apart).

You have the whole class period (75 minutes) to complete this exam.

Points will not be counted off for small syntax errors.

Part I (16 pts. total)

1. Convert the following numbers to the other formats (2 pts. each)

a. Decimal 27 Binary 11011 Hex 0x1b

b. Decimal 21 Binary 10101 Hex 0x15

c. Decimal 28 Binary 11100 Hex 0x1c

d. Decimal 15 Binary 1111 Hex 0xf

Part II (84 pts. total)

2. (2) What are two ways to write comments in Java programs?

```
/* comment */  
// comment
```

3. (3) Consider this example code.

```
final int i = 100;  
  
while(i > 0){  
    System.out.println(i);  
    i = i - 10;  
}
```

- a. Is it legal Java (yes or no)? **NO**
- b. Why or why not?

A final variable cannot be assigned a new value.

4. (3) Consider the following program fragment.

```
Public class Class1  
{  
    static int a = 0;  
  
    public static void main()  
    {  
        int a = 1;  
  
        System.out.println("a = " + a);  
    }  
}
```

What is printed?

a = 1

5. (3) Consider the following program fragment.

```
int i = 100;  
i = 30 / 10 + 5;
```

What is the value of `i` after the program fragment executes? 8

6. (8) Which of the following are legal Java code fragments? Circle true or false.

a. Legal (true or false) false

```
int i = 0;  
double d = 0;  
i = d;
```

b. Legal (true or false) true

```
int i = 0;  
double d = 0;  
d = i;
```

c. Legal (true or false) false

```
int i = 0;  
byte b = 0;  
b = i;
```

d. Legal (true or false) true

```
int i = 0;  
byte b = 0;  
i = b;
```

7. (2) Consider the following program fragment.

```
int i;  
i = (int) 13.4;
```

What is the value of `i` after the program fragment executes? 13

8. (2) Consider the following program fragment.

```
int i = 0;
int z = 0;

if(z < 0 && i++ == 0)
    i = 3;
System.out.println(i);
```

What number is printed? 0

9. (3) What will this program fragment print?

```
char c = 'a';

switch(c){
case 'a':
    System.out.println("It was a");

case 'c':
    System.out.println("It was c");

default:
    System.out.println("Neither a nor c");
}
```

It was a

It was c

Neither a nor c

10. (4) What will this program fragment print?

```
for(int i = 0; i < 20; i = i + 10) {
    if (i < 10)
        continue;
    System.out.println(i);
}
```

11. (3) What will this program fragment print?

```
int a[] = new int[2];
int b[] = new int[2];

a[0] = 1;
a[1] = 2;
b[0] = 1;
b[1] = 2;
if(a == b)
    System.out.println("equal");
else
    System.out.println("not equal");
```

not equal

12. (3) What will this program print?

```
public class Class1
{
    public static void change(int n)
    {
        n = 3;
    }

    public static void main (String[] args)
    {
        int n = 2;

        System.out.println("n = " + n);
        change(n);
        System.out.println("n = " + n);
    }
}
```

n = 2
n = 2

13. (4) Which of the following operations on a Java reference are possible? Please circle the ones that are legal in Java.

- a. Setting the value with =. **LEGAL**
- b. Adding to it with +.
- c. Invoking a method on a referenced object. **LEGAL**
- d. Passing the value of the reference as a parameter to a method. **LEGAL**

Name _____ TA: Han [] Sherman []
Cooper [] Yang []

14. (2) Is this legal Java (yes or no)? **NO**

```
int n = new int;
```

15. (4)

a. What is garbage collection?

Reclaiming unused memory.

b. When does it occur?

After the memory is no longer referenced.

When a reference no longer points to the memory.

Or something similar.

16. (3) Please label these as showing an example of a formal, an actual, or neither (circle one).

a. (formal, actual, neither) **neither**

```
z = a++;
```

b. (formal, actual, neither) **actual**

```
System.out.println( a );
```

c. (formal, actual, neither) **formal**

```
public Employee(String theName)  
{  
}
```

17. (2) Can you access a private instance variable of a superclass from a method in a subclass (yes or no)? **NO**

18. (2) Can a constructor return a value (yes or no)? If yes, what might it return?

NO is what I meant. However, you could say that it returns a reference to new object.

19. (2) Is every class *you write* in Java a subclass (yes or no)?

Yes, subclass of Object.

Name_____ TA: Han [] Sherman []
Cooper [] Yang []

20. (2) Why would you write a static method?

To execute some function that does not use/need any class variables.

21. (2) What is the use for a static class variable?

A single copy of some value associated with the class.

22. (2) What does the instanceof operator do?

Tests whether the referenced object on the left is of the class listed on the right.

Results in Boolean value.

23. (3) Can you keep a superclass constructor from executing (yes or no)? NO

24. (5) Write code that will print all of the command-line arguments of main(), one per line.

```
public class Class1
{
    public static void main (String[] args)
    {
        for(int i = 0; i < args.length; i++)
            System.out.println(args[i]);
    }
}
```

This is from the Command Line example I showed in class (it's on class web page).

Name_____ TA: Han [] Sherman []
Cooper [] Yang []

25. (4) What is printed by the following example?

I forgot to
type "static".

```
static public class Class1
{
    public void main(String args[])
    {
        Employee p = new Employee("Joe", 10000.00);
    }
}
```

Constructor of Person No Name
Employee Constructor 10000.0

In file Employee.java:

```
public class Employee extends Person
{
    public Employee(String theName, double sal)
    {
        salary = sal;
        System.out.println("Employee Constructor " + salary);
    }

    private double salary = 0;
}
```

In file Person.java:

```
public class Person {
    public Person(String theName, int sal)
    {
        name = theName;
        System.out.println("Constructor of Person " + name);
    }

    public Person(String theName)
    {
        name = theName;
        System.out.println("Constructor of Person " + name);
    }

    public Person()
    {
        name = "No Name";
        System.out.println("Constructor of Person " + name);
    }

    private String name = null;
}
```

26. (3) In the context of the code in the previous question

- a. Would this be a legal Java replacement for the Class1 of the previous example (yes or no)? **YES (except that I forgot static)**

```
public class Class1
{
    static public void main(String args[])
    {
        Person [ ] p = new Person[ 4 ];

        p[0] = new Person( "joe");
        p[1] = new Employee( "bob", 1000000);
    }
}
```

- b. Explain why or why not.

**Answer I was looking for is that a reference of the superclass can reference the subclass.
An acceptable answer is NO because there is no static modifier.**

27. (2) Is it possible to have a private method? (yes or no) **YES**

28. (3) I am creating a program that stores information about the students and staff at UNC. I propose to have a class Person to store information about each of the people. One of the items that I wish to store is the address, either home or campus.

- a. Should I create a class Address and make it a subclass of Person (yes or no)?

NO

- b. Why or why not?

Address does not have an Is-A relationship to Person. The relationship is a Has-A relationship.

29. (4) What is encapsulation and why would we want to use it in programming?

Encapsulation is the practice of hiding the internal details and only exposing functionality. Could also be stated as hiding the data and only exposing through methods.

You use it to make programs easier to update.