

James Finn

CPE 231 2CD

1 October, 2009. 13:00-16:00

Programming Language Seat Number: _____

¹
Final Examination

Name _____

ID # _____

General Instructions


This exam is multiple choice. There are 20 questions. Each question has four possible answers, labeled *a* through *d*. Each correct answer is worth 5 points, for a total of 100 points possible. Each incorrect or blank answer is scored 0 points

Write all of your answers directly on the answer sheet on the last page.

The examination is open book. You may make use of books, handouts, notes and a dictionary. You may not use electronic devices of any kind, including telephones, pagers, computers or electronic calculators.

Make sure your exam is not missing any pages. There are 16 pages total.

Instructor

 _____

(Dr. James Finn)

Problem 1.

Which of the following statements about encapsulation are true?

1. Java supports encapsulation.
2. C++ supports encapsulation.
3. C supports encapsulation.
4. C does not support encapsulation, but it is possible to implement encapsulation through the use of incomplete types.

Answers:

- (a) 1, 2 and 3 are true.
- (b) 1 and 2 are true.
- (c) 1 and 4 are true.
- (d) 1, 2 and 4 are true.

Problem 2.

Java is not a “pure” object-oriented language. It has objects, but it also has *primitive types* such as **int** and **char**. Which of the following are true?

1. An array is a primitive type.
2. A **String** is a primitive type.
3. Java treats primitive types differently from objects.
4. All objects are subclasses of the Java class **Object**.

Answers:

- (a) 1, 3 and 4 are true.
- (b) 3 and 4 are true.
- (c) All of the statements are true.
- (d) 1, 2 and 3 are true.

Problem 3.

In C, objects allocated from the free store (by **malloc** or **calloc**) remain allocated until they are freed by a call to **free**. In Java, objects are allocated from the free store using the keyword **new**. Java has no equivalent to C's **free**. In Java, objects are *garbage collected*, meaning the runtime system tracks their usage and frees them when they are no longer in use.

Which of the following are true?

1. It is easier to program with garbage collection than it is to program with a manual memory management system like C's.
2. Garbage collection prevents all memory leaks.
3. Garbage collection used to be very inefficient, but modern garbage collection algorithms are competitive with manual memory management.

Answers:

- (a) 1 and 3 are true.
- (b) 1 and 2 are true.
- (c) All of the statements are true.
- (d) All of the statements are false.

Problem 4.

What will be the output when the following **main** program is run?

```
public class StaticTest
{
    private static int track = 0;
    private int local = 0;

    public static int getTrack() { return track; }
    public int getLocal() { return local; }

    public static void f(int arg)
    {
        track = track + arg;
    }

    public void in(int arg)
    {
        local = local + arg;
        track = track + local;
    }

    public static void main(String args[])
    {
        StaticTest a = new StaticTest();
        StaticTest b = new StaticTest();
        a.in(3);
        StaticTest.f(4);
        b.in(3);
        StaticTest.f(4);
        System.out.println("a local = " + a.getLocal());
        System.out.println("b local = " + b.getLocal());
        System.out.println("track = " + StaticTest.getTrack());
    }
}
```

- (a)
a local = 3
b local = 3
track = 14
- (b)
a local = 3
b local = 3
track = 7
- (c)
a local = 3
b local = 6
track = 17
- (d)
a local = 3
b local = 3
track = 4

Problem 5.

In Java, you can create a class *B* that is a subclass of class *A* by declaring that class *B* **extends** *A*. Which of the following are true?

1. Class *B* can declare new instance data.
2. Class *B* can declare new instance methods.
3. Class *B* can override instance methods inherited from *A*.
4. Class *B* can override constructors from class *A*.
5. Class *B* can remove methods that are inherited from *A*.

Answers:

- (a) All of the statements are true.
- (b) 1, 2, 3 and 4 are true.
- (c) 1, 2 and 3 are true.
- (d) 1, 2, 3 and 5 are true.

Problem 6.

Which of the following are true about subclasses and subtypes?

1. Every subclass is always a subtype.
2. In Java, the programmer should create a subclass only if it is also a subtype.
3. The Java compiler can check whether a subclass is a subtype.

Answers:

- (a) Only statement 3 is true.
- (b) All of these statements are true.
- (c) 1 and 2 are true.
- (d) Only statement 2 is true.

Problem 7.

Given these two Java classes:

```
public class PolyBase
{
    protected int x;

    PolyBase() { x = 0; }

    public int f()
    {
        x++;
        return x;
    }

    public int g()
    {
        return f() + 2;
    }

    public int h()
    {
        return g() + 1;
    }
}
```

```
public class PolySub extends
PolyBase
{
    public int f()
    {
        x = x + 3;
        return x;
    }

    public int h()
    {
        return f();
    }
}
```

What will be the output when the following **main** program is run?

```
public static void main(String[] args)
{
    int total = 0;
    PolyBase pb = new PolyBase();
    total = total + pb.f();
    total = total + pb.g();
    total = total + pb.h();
    pb = new PolySub();
    total = total + pb.f();
    total = total + pb.g();
    total = total + pb.h();
    System.out.println("total = " + total);
}
```

Answers:

- (a) total = 0
- (b) total = 27
- (c) total = 29
- (d) total = 31

Problem 8.

What is the value of the function call $f(5, 3)$?

```
public static int f(int a, int b)
{
    if (b == 0) return 0;
    else return a + f(a, b-1);
}
```

Answers:

- (a) 20
- (b) undefined, error in recursion
- (c) 0
- (d) 15

Problem 9.

Which of the following statements about the function f from Problem 8 are true?

1. f is tail recursive.
2. The base case in f is when the second argument b is equal to 0.
3. This function will fail because of infinite recursion if the second argument b is negative.
4. If the second argument b is not negative, the value of $f(a, b)$ is $a*b$.

Answers:

- (a) 2, 3 and 4 are true.
- (b) 2 and 3 are true.
- (c) 3 and 4 are true.
- (d) All of the statements are true.

Problem 10.

The Scheme programming language has both *functions* and *special forms*. Consider the expression

```
(fsf exp1 exp2 exp3)
```

where **fsf** is a function or a special form, and **exp1**, **exp2** and **exp3** are Scheme expressions. Which of the following statements are true?

1. If **fsf** is a special form, some of the expressions **exp1**, **exp2** and **exp3** may not be evaluated.
2. If **fsf** is function, all of the arguments **exp1**, **exp2** and **exp3** will be evaluated before the function is called.
3. If **fsf** is a special form, the form controls in what order the argument expressions are evaluated.

Answers:

- (a) 1 and 3 are true.
- (b) 1 and 2 are true.
- (c) None of the statements are true.
- (d) All of the statements are true.

Problem 11.

Given these definitions:

```
(define (g f x)
  (f x x))
```

```
(define (conser a b)
  (cons a b))
```

What is the value of the expression

```
(g conser '(a b))
```

Answers:

- (a) (conser '(a b) '(a b))
- (b) ((a b) a b)
- (c) (a b a b)
- (d) ((a b) (a b))

Problem 12.

Given these definitions:

```
(define x 'red)
(define y '(a b c))
```

Which of the following Scheme expressions are true (evaluate to #t)?

1. (eq? 'red 'red)
2. (eq? x 'red)
3. (eq? y (cons 'a '(b c)))
4. (equal? x 'red)
5. (equal? y (cons 'a '(b c)))

Answers:

- (a) All of the expressions are true.
- (b) 1 and 2 are true.
- (c) 1, 2, 4 and 5 are true.
- (d) 1, 2 and 5 are true.

Problem 13.

Given this function definition:

```
(define (lister x)
  (cond ((null? x) '(a))
        ((pair? x) (cons 'a (lister (cdr x))))
        (else x)))
```

What is the value of the expression

```
(lister '(x y))
```

Answers:

- (a) (a x y)
- (b) (a a)
- (c) (a a a)
- (d) (a)

Problem 14.

Which of the following Scheme expressions are true?

1. `(equal? '(5 5) (cons 5 '(5)))`
2. `(equal? '(5 5) (cons 5 5))`
3. `(equal? '(5 . 5) (cons 5 5))`
4. `(equal? '(5 5) (cons 5 (cons 5 '())))`

Answers:

- (a) 1 and 3 are true.
- (b) 1, 2 and 4 are true.
- (c) All of the statements are true.
- (d) 1, 3 and 4 are true.

Problem 15.

Given these Scheme definitions:

```
(define (constructor)
  (define s 0)
  (define (dispatch op data)
    (cond ((eq? op '+) (set! s (+ s data)))
          ((eq? op 'get) s)))
  dispatch)

(define (store dispatch x) (dispatch '+ x))
(define (retrieve dispatch) (dispatch 'get 'ignore))
```

We evaluate these expressions, in order:

```
(define test (constructor))
(store test 5)
(store test 6)
(store test 3)
(retrieve test)
```

What is the value of the last expression `(retrieve test)`?

Answers:

- (a) `#<procedure:dispatch>`
- (b) 5
- (c) 14
- (d) 3

Problem 16.

Traditional means of reporting errors and exceptional conditions include functions returning special *error return values* and the use of error *callback functions*. *Exception handling* is a modern programming language feature designed to eliminate the problems associated with traditional techniques.

Which of the following statements are true?

1. It is not always possible to use special error return values, because for many functions there is no special return value available that could mean that an error was detected.
2. A special error return value may not reach the part of the client code that is prepared to handle the error.
3. If an implementation detects an error and calls a client callback function, the implementation may not know how to continue if the callback function returns.
4. Exception handling is a good solution to the problem of error handling because the function that throws the exception does not have to continue executing and the exception can be caught in the part of the client code that is best prepared to handle the error.

Answers:

- (a) None of the statements are true.
- (b) All of the statements are true.
- (c) 1, 2 and 4 are true.
- (d) 1, 3 and 4 are true.

Problem 17.

What will be the output when this Java `main` program is run?

```
public class ExceptTest
{
    public void a()
    {
        try
        {
            b(null);
            f(null);
        }
        catch (NullPointerException e)
        {
            System.out.println(
                "a caught a NullPointerException");
        }
    }

    public void b(Object x)
    {
        try
        {
            f(x);
        }
        catch (NullPointerException e)
        {
            System.out.println(
                "b caught a NullPointerException");
        }
    }

    public void f(Object arg)
    {
        if (arg == null)
            throw new NullPointerException("null pointer");
    }

    public static void main(String args[])
    {
        try
        {
            ExceptTest et = new ExceptTest();
            et.f("hello");
            et.a();
            et.b("hello");
        }
        catch (NullPointerException e)
        {
            System.out.println(
                "main caught a NullPointerException");
        }
    }
}
```

Problem continues on next page:

Answers:

- (a)
 - main caught a NullPointerException
 - b caught a NullPointerException
 - a caught a NullPointerException
 - b caught a NullPointerException
- (b)
 - main caught a NullPointerException
- (c)
 - a caught a NullPointerException
 - a caught a NullPointerException
- (d)
 - b caught a NullPointerException
 - a caught a NullPointerException

Problem 18.

Consider these Java classes:

```
public class E1 extends Exception
{
    public E1(String msg)
    {
        super(msg);
    }
}

public class E2 extends RuntimeException
{
    public E2(String msg)
    {
        super(msg);
    }
}
```

Problem continues on next page:

```

public class TestExcept
{
    public void f()
    {
        throw new E1("error");
    }

    public void g()
    {
        throw new E2("error");
    }

    public void test1()
    {
        f();
        System.out.println("f returned successfully");
    }

    public void test2()
    {
        g();
        System.out.println("g returned successfully");
    }
}

```

Which of the following statements are true?

1. Function **f** will not compile because it does not declare that it **throws E1**.
2. Function **g** will not compile because it does not declare that it **throws E2**.
3. Function **test1** will not compile because it does not catch **E1** and it also does not declare that it **throws E1**.
4. Function **test2** will not compile because it does not catch **E2** and it also does not declare that it **throws E2**.

Answers:

- (a) None of the statements are true.
- (b) Statements 1 and 3 are true
- (c) Statements 2 and 4 are true.
- (d) All of the statements are true.

Problem 19.

Consider the following PHP code:

```
The person <?php echo htmlspecialchars($_POST['name']); ?>  
is too young to visit this Web site. Try disney.com.
```

Which of the following is true?

- (a) This code is an error because you cannot put PHP tags inside of normal HTML text.
- (b) This code might be used to respond to submission of a form.
- (c) The function **htmlspecialchars** formats the name in bold font to look “special”.
- (d) **\$_POST** means that this is part of the Bangkok Post Web site.

Problem 20.

Which of the following statements about PHP arrays are true?

- 1. You can use nonnegative integers as array indices. For example: **`$a[1] = 3.`**
- 2. You can use strings as array indices. For example: **`$a["Harry"] = "Potter".`**
- 3. When you use strings as array indices, there is no way to iterate over the array values.

Answers:

- (a) All of the statements are true.
- (b) None of the statements are true.
- (c) Statement 1 is true.
- (d) Statements 1 and 2 are true.