CSE 341 Midterm — May 11, 1998

Your name:

1. (7 pts) Suppose the following Scheme code has been evaluated:

```
(define xsq 100)
(define ysq 200)

(define (square x)
    (* x x))

(define (hypotenuse x y)
   (let ((xsq (square x))
         (ysq (square y)))
      (sqrt (+ xsq ysq))))

(define a '(fred ethel ricky lucy))
(define b a)
(set-car! a 'sam)
(set! a 100)
```

Then what is the result of evaluating the following Scheme expressions? If there is an error say so. Write your answer using the printable representation of the result (i.e., what the Scheme system outputs after evaluation).

(a) `(hypotenuse 3 4)`

(b) `(map square '(1 2 3 4))`

(c) `xsq`

(d) `(map (lambda (a) (cons a ())) '(1 2 3))`

(e) `a`

(f) `b`

(g) `(caddr b)`

2. (8 pts) Suppose that we evaluate the following Smalltalk code.

```
| a b c |
a := Array new: 2.
b := Array new: 2.
c := a.

a at: 1 put: 10.
a at: 2 put: 20.

b at: 1 put: 30.
b at: 2 put: 20.

a=b ifTrue: [Transcript show: 'a and b are equal'.  Transcript cr].
a==b ifTrue: [Transcript show: 'a and b are identical'.  Transcript cr].
a=c ifTrue: [Transcript show: 'a and c are equal'.  Transcript cr].
a==c ifTrue: [Transcript show: 'a and c are identical'.  Transcript cr].

Transcript show: 'changing an element ...'.  Transcript cr.
c at: 1 put: 30.

a=b ifTrue: [Transcript show: 'a and b are equal'.  Transcript cr].
a==b ifTrue: [Transcript show: 'a and b are identical'.  Transcript cr].
a=c ifTrue: [Transcript show: 'a and c are equal'.  Transcript cr].
a==c ifTrue: [Transcript show: 'a and c are identical'.  Transcript cr].
```

What is printed to the transcript?

3. (9 pts) Suppose we are simulating trains in Smalltalk, and define a class Engine, as follows.

```
Object subclass: #Engine
  instanceVariableNames: 'idNumber'
  classVariableNames: ''
  poolDictionaries: ''


setIdNumber: n
  "set the engine's identification number (each engine should have one)"
  idNumber := n

kind
  "return a descriptive string saying what kind of engine this is"
  ^ 'engine'

printOn: aStream
  aStream nextPutAll: self kind.
  idNumber printOn: aStream.
```

One subclass of Engine is DieselEngine, defined as follows:

```
Engine subclass: #DieselEngine
  instanceVariableNames: ''
  classVariableNames: ''
  poolDictionaries: ''

kind
  ^ 'diesel locomotive '
```

Another subclass of Engine is SteamEngine, defined as follows:

```
Engine subclass: #SteamEngine
  instanceVariableNames: ''
  classVariableNames: ''
  poolDictionaries: ''

kind
  " , (comma) is the string concatenation message "
  ^ 'steam ' , super kind
```

Suppose we have made instances of the three classes as follows:

```
a := Engine new.       a setIdNumber: 8.
b := DieselEngine new. b setIdNumber: 17.
c := SteamEngine new.  c setIdNumber: 97.
```

What is printed to the transcript for each of the following messages?

```
a printOn: Transcript.
b printOn: Transcript.
c printOn: Transcript.     'typo in original exam fixed'
```

Briefly explain why Smalltalk gives these responses.

4. (5 pts) Continuing with the railroad theme, suppose we have defined the following classes: Train, Tracks, FreightTrain, GenericRailroadCar, PassengerTrain, Engine, PassengerCar, and BoxCar. Which of these should be subclasses of which? Draw the subclassing graph with edges pointing up from subclasses to their superclasses (be sure to include all of the classes in your diagram).

5. (8 pts) Java has two keywords it uses to specify two different kinds of inheritance: `extends` and `implements`. Explain the major differences between these two kinds of inheritance relationships. Define “multiple inheritance,” and relate your discussion to that concept.

6. (8 pts) What are the tradeoffs between using languages with garbage collection, with reference counting, and programmer-controlled deallocation?

7. (9 pts) Suppose another programmer has defined the Java public class ColoredBall which extends the public class Ball (i.e., ColoredBall **is-a** Ball). Both classes define a `bounce()` method. Consider the below method, `keepBouncing()` for class Ball:

```
public class Ball
  {
  // ... other code here ...
  void keepBouncing() { while(true) { this.bounce(); } }
  // ... other code here ...
  }
```

(a) Does ColoredBall's or Ball's bounce() method get invoked from `keepBouncing()`? Explain.

(b) What is the "this" in `this.bounce()`? What is it used for? Is it necessary?

(c) Suppose class Ball also defines another method like so:

```
public class Ball
  {
  // ... other code here, including method from above ...
  void keepBouncing(int i) { /* some body for this method */ }
  }
```

does this new method *override* or *overload* the first `keepBouncing` method? What is the diffence between those two terms? Give an example of the other (e.g., if this is an example of overriding, give an example of overloading, or vice-versa).

8. (8 pts) Consider the following Smalltalk class definition:

```
Object subclass: #Squid
  instanceVariableNames: ''
  classVariableNames: ''
  poolDictionaries: ''

test: n block: aBlock
  Transcript show: 'n = '; show: n printString; cr.
  Transcript show: 'value = '; show: aBlock value printString; cr.
  n < 3 ifTrue: [self test: n+1 block: [n]]
```

What is printed to the transcript if we evaluate

```
   | s |
 s := Squid new.
 s test: 1 block: [20+30]
```