CSE 3302   Name _____
Test 1
Fall 2014
Multiple Choice. Write your answer to the LEFT of each problem. 4 points each
1.   Which celebrity has not received an ACM Turing Award?
     A. Alan Kay B. John McCarthy   C. Dennis Ritchie   D. Bjarne Stroustrup
2.   Short-circuit evaluation has historically been missing from which language?
           A. C   B. JavaScript   C. Pascal   D. Scheme
3.   Which of the following is not emphasized in *Elements of Programming*?
     A. Behavioral vs. representational equality
     B. Class/subclass hierarchies
     C. Decomposition and rigorous design decisions
     D. Separation of algorithms and data structures
4.   It is considered good practice to assure that the second argument to `cons` is:
           A. an atom   B. a function   C. a list   D. a number
5.   Which language has the smallest set of first class "objects"?
           A. C++   B. JavaScript   C. PL/0   D. Scheme
6.   For C++, what language feature is preferred over macros?
     A. Built-in strings without null terminators   B. Operator overloading
     C. Resource allocation is initialization (RAII)   D. Templates
7.   Which of the following is true regarding attribute grammars?
     A. Inherited attributes carry information up the parse tree
     B. Synthesized attributes carry information down the parse tree
     C. They can represent context-sensitive information
     D. They cannot capture the information that usually goes in symbol tables
8.   Lab 1 required modifying which part of PL/0?
           A. Hand-written scanner   B. Railroad diagrams
           C. Recursive-descent parser/code generator   D. Stack-based interpreter
9.   The necessity for programming languages to evolve is advocated by:
           A. Edsger Dijkstra   B. John McCarthy   C. Guy Steele   D. Bjarne Stroustrup
10. Many development organizations require the use of `{` and `}` when coding control structures in a C derivative language.
     This avoids which issue?
           A. dangling else   B. subscripts out of range
           C. unmatched delimiters   D. exceptions
11. Which language does not allow nesting functions?
           A. C++ B. Scheme   C. JavaScript   D. Pascal
12. What is the result of
     `(cdr (cons (car '(a (b c (d e)) f g)) (cdr '((h i) j (k l)))))` ?
     A. `'(j (k l))`   B. `'(a j (k l))`
     C. `'((b c (d e)) f g)`   D. `'(h i)`
13. What is the result of `(and 'a '(b c) 'd)` ?
     A. `'(a (b c) d)` B. `'d`   C. `'a`   D. `#t`
14. Regular expressions are convenient for defining what part of a programming language?
           A. attributes   B. binary strings C. tokens   D. terminals
15. In C++, what operator is overloaded to provide simple output?
           A. `&`   B. `+=`   C. `<<`   D. `cout`
Long Answer.
1.   Give a Scheme predicate `test1?` that indicates whether its three integer parameters are all positive. 10 points
     The results for a few specific cases below are indicated.
     `(test1? 1 2 3)` should return `#t`
     `(test1? 0 -1 -2)` should return `#f`
     `(test1? 3 2 0)` should return `#f`
2.   Give a JavaScript function `test1` that indicates whether its three integer parameters are all positive. The result should be
     returned to the caller. 10 points
3.   Give a JavaScript function `allEqual` that takes an array of integers as its only argument and returns a Boolean value to
     indicate whether all of the values are the same. 20 points
     The results for a few specific cases below are indicated.
     `allEqual ([1,1,1,1])` should return `true`
     `allEqual ([1,1,2,1])` should return `false`

```
allEqual ([2])  should return true
allEqual ([])  should return true
allEqual ([-1,1,1,1]) should return false
```

CSE 3302   Name _____

Test 2

Fall 2014

Multiple Choice. Write your answer to the LEFT of each problem. 5 points each
1.   Which is true about the run-time cost of processing a thrown exception?
   A. It is proportional to the number of dynamic links followed
   B. It is always constant
   C. It depends on the numbers of handlers and dynamic links encountered
   D. It is proportional to the number of static links followed
2.   Suppose the C declaration below occurs at global scope. Where would the space be allocated?
   `int arr[10000];`
   A. static B. heap C. stack D. registers
3.   PL/0 uses static links to:
   A. Update the display table   B. Return from a called procedure
   C. Place an integer on the stack   D. Reference data
4.   Suppose a variable is referenced in a subroutine closure. Where is it stored?
   A. static B. heap C. stack D. registers
5.   The notion of l-value and r-value is associated with which PL construct?
   A. assignment   B. iteration   C. recursion   D. selection
6.   Duff's device involves which PL construct?
   A. C union   B. C switch   C. Java switch   D. C varargs
7.   Which language does not allow nesting functions?
   A. C   B. Scheme   C. JavaScript   D. Pascal
8.   For JavaScript, which expression always gives the same value as `a || b`?
   A. `!(!a || !b)`   B. `a ? b : a` C. `a ? a : b` D. `b && a`
9.   "Hoisting" of declarations to the beginning of functions is associated with which language?
   A. C   B. JavaScript   C. Scheme   D. Pascal
10. Jensen's Device implements a higher-order procedure using:
   A. call by name   B. functional arguments   C. static links   D. call by value
11. C++ controls name conflicts by using:
   A. `&`   B. exception hierarchies   C. namespaces   D. overloading
12. Which of the following binding times is the latest?
   A. compilation   B. execution   C. linking   D. program writing

Long Answer.
1.   Give *equivalent inline* C code (e.g. using `if … else …`) to demonstrate the short-circuit nature of C boolean operators. Do not use `&&`, `||`, or `!` in your solution! Do not use work variables! Do not use `return`! (20 points)
   a. `result = a > 10 && b < 13;`
   b. `result = c > 20 || d <= 17;`
   c. `result = !(e > 25 || f < 55) && g > 66;`
2.   Write a Scheme function (along with helper functions) that takes an unordered list of integers (with at least one atom) and moves the smallest value to the end of the list. If the smallest value appears several times, then the *last* occurence (going left-to-right) will be the one moved to the end. 20 points
   ```
   > (smallToEnd '(1 2 2 1 2 2 1 2))
   '(1 2 2 1 2 2 2 1)
   > (smallToEnd '(11 22 22 11 22 22 33 22 33))
   '(11 22 22 22 22 33 22 33 11)
   > (smallToEnd '(1))
   '(1)
   > (smallToEnd '(1 2 3 4))
   '(2 3 4 1)
   > (smallToEnd '(4 3 2 0 2 3 4))
   '(4 3 2 2 3 4 0)
   ```

CSE 3302   Name _____

Test 3

Fall 2014

Multiple Choice. Write your answer to the LEFT of each problem. 4 points each
1.   The notion of a mixin is associated with?

      A. ML   B. C++ templates   C. JavaScript   dynamic typing D. multiple inheritance

2. Existential types for abstract data types allow multiple
      A. operations   B. implementations   C. interfaces   D. inheritance

3. A new property may be added to a JavaScript object using:
      A. assignment   B. `new` C. `Object.create()` D. prototypal inheritance

4. The `for .. in` construct is used to:
      A. enumerate array elements that are not `undefined`   B. enumerate properties
      C. test a subclass/superclass relationship   D. iterate over an integer subrange

5. Which of the following JavaScript objects does not have a `length`?
      A. arrays   B. functions   C. numbers   D. strings

6. Stop-and-copy is an example of:
      A. reference counts   B. garbage collection
      C. deep comparison   D. run-time stack implementation

7. Indicate the value of
```
((lambda (x y)
   (x (x y)))
 (lambda (z) (+ 5 z))
 10)
```
      A. 5   B. 10   C. 15   D. 20

8. Which of the following will be treated like `false` (In JavaScript)?
      A. 5   B. `undefined`   C. `"else"`   D. `"false"`

9. Dynamic binding in a statically-typed language is exemplified by
      A. JavaScript   B. Scheme   C. Java D. Pascal

10. The state of a Scheme computation may be saved as:
      A. a closure   B. a continuation C. a combinator D. a thread

Long Answer.

1. What is the result of executing this Scheme code? (10 points)
```
(define (f x y) (* 5 (+ x y)))
((lambda (y x z)
   (f y (x y z)))
 10
 *
 3)
```

2. What appears on the console for the code below? (10 points)
```
function g(x) {
  return {
    a: function (y) {
      x.push(y);
    },
    b: function () {
      return x.pop();
    },
    c: function () {
      return x.length;
    },
    d: function(sub,val) {
      x[sub]=val;
    }
  };
}
var a=g([5,3,2,4]);
a.d(3,7);
console.log(a.b());
a.a(11);
a.a(22);
a.d("Romo",9);
console.log(a.c());
console.log(a.x);
```

3. The JavaScript code below was provided along with lab 4. Carefully indicate the additional code to give each of the three classes a `min()` method to give the minimum value for that class (20 points)

    a.   For the `pointClass`, the minimum is the smallest `v` provided to the constructor
    b.   For the `intervalClass`, the minimum is the smallest `l` provided to the constructor
    c.   For the `rootClass`, the minimum is the smallest value over the two subclasses

```javascript
var rootClass,intervalClass,pointClass;

rootClass=(function () { // just counts total number of instances

  var instanceCount=0;

  return {

    add1:  function () {instanceCount++;},

    count: function () {return instanceCount;}

   };

  })();
intervalClass=Object.create(rootClass);

var wrkI=(function () {

            var instanceCount=0; // access in closure

            return {

                  construct: function (l,r) {

                              // Each instance has intervalClass as prototype

                              if (l>r) {

                                 lab4output.innerHTML+=

                                   "invalid call to construct "+l+" "+r;

                                 throw "invalid call to construct "+l+" "+r;

                                 }

                              this.add1(); // for total number of objects

                              instanceCount++;

                              var obj=Object.create(intervalClass);

                              obj.get=function () {

                                        return {left:l,right:r};

                                        };

                              return obj;

                              },

                  count: function () {

                            return instanceCount;

                            }

                };

            })();
intervalClass.construct=wrkI.construct;

intervalClass.count=wrkI.count;

intervalClass.inside=function (x) {

                        var work=this.get();

                        return work.left<=x && x<=work.right;

                        };
intervalClass.measure=function () {
```

```
                    var work=this.get();
                    return work.right-work.left;
                    };
pointClass=Object.create(rootClass);
var wrkP=(function () {
        var instanceCount=0; // access in closure
        return {
            construct: function (v) {
                    // Each instance has intervalClass as prototype
                    this.add1(); // for total number of objects
                    instanceCount++;
                    var obj=Object.create(pointClass);
                    obj.get=function () {
                            return v;
                            };
                    return obj;
                    },
            count: function () {
                    return instanceCount;
                    }
        };
    })();
pointClass.construct=wrkP.construct;
pointClass.count=wrkP.count;
pointClass.inside=function (x) {
                return this.get()==x;
                }
pointClass.measure=function () {return 0;};
var a=intervalClass.construct(20,30);
var b=intervalClass.construct(25,35);
var c=intervalClass.construct(7,90);
var d=pointClass.construct(100);
var e=pointClass.construct(50);
var f=pointClass.construct(70);
console.log(rootClass.min());      // Gives 7
console.log(intervalClass.min())   // Gives 7
console.log(pointClass.min());     // Gives 50
```

4. Suppose a Pascal array is to be stored starting at location 20000 and is declared:
   c: array[24..70,20..33,5..8] of integer;
   If one integer takes two bytes, what is the location of c[44,22,7]? (20 points)