CSE 3302 Name ______________________________
Test 1
Fall 2013

Multiple Choice. Write your answer to the LEFT of each problem. 4 points each

1. If a value is second class, then it can be
   A. Assigned to a variable B. Returned from a function
   C. Passed as an argument D. All of the above
2. Which language was developed most recently?
   A. Scheme B. JavaScript C. Pascal D. C
3. Which language does not allow nesting functions?
   A. C B. Scheme C. JavaScript D. Pascal
4. `(car (cdr (car (cdr '(a (b (c d e) f (g h) i))))))` will result in:
   A. `’((g h i))` B. `’b` C. `’(c d e)` D. `’(g h i)`
5. A C compiler does which phase first?
   A. scanning B. preprocessor C. parsing D. semantic analysis
6. Which language’s operator precedences bear the least similarity to the other three?
   A. Pascal B. Java C. JavaScript D. C
7. Which of the following is not a JavaScript feature?
   A. first-class functions B. strong type checking
   C. event-driven execution D. integration with HTML
8. Which language does not evaluate boolean expressions with “short circuiting”?
   A. Scheme B. JavaScript C. Pascal D. C
9. Who is associated with the Pascal language?
   A. Dijkstra B. Hoare C. Ritchie D. Wirth
10. Which of the following allows anonymous functions?
    A. C B. Scheme C. PL/0 D. Pascal

Long Answer.

1. Write a Pascal function `test1` to determine the median (second largest) of its three integer parameters. You may assume the three provided values are unique. 20 points
   All six of the following should return `20`:
```
test1(10,20,30);
test1(20,10,30);
test1(10,30,20);
test1(30,10,20);
test1(20,30,10);
test1(30,20,10);
```
2. Write a Scheme function `test1` to determine the median (second largest) of its three integer parameters. You may assume the three provided values are unique. 20 points
   All six of the following should return `20`:
```
(test1 10 20 30)
(test1 20 10 30)
(test1 10 30 20)
(test1 30 10 20)
(test1 20 30 10)
(test1 30 20 10)
```
3. Write a JavaScript function `test1` to determine the median (second largest) of its three integer parameters. You may assume the three provided values are unique. 20 points
   All six of the following should return `20`:
```
test1(10,20,30);
test1(20,10,30);
test1(10,30,20);
test1(30,10,20);
test1(20,30,10);
test1(30,20,10);
```

CSE 3302 Name ______________________________
Test 2
Fall 2013

Multiple Choice. Write your answer to the LEFT of each problem. 4 points each

1. Which of the following has no mechanism for achieving block scope?
   A. C B. JavaScript C. Scheme D. Pascal

2. Suppose the C declaration below occurs at global scope. Where would the space be allocated?

```
int arr[10000];
```

   A. staticB. heap C. stackD. registers

3. 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

4. Suppose a variable is referenced in a subroutine closure. Where is it stored?

   A. staticB. heap C. stackD. registers

5. Who invented attribute grammars?

   A. Dijkstra B. Hoare C. Knuth D. Wirth

6. Lookahead is a concept that is important for:

   A. attribute evaluation B. parsing C. operational semantics D. Top-down

7. Which of the following is true regarding attribute grammars?

   A. Synthesized attributes carry information down the parse tree
   B. Inherited attributes carry information up the parse tree
   C. They cannot represent context-sensitive information
   D. They are not used for constructing scanners

8. Which of the following is never stored on the stack for the shunting-yard technique?

   A. (B.) C. unary operators D. binary operators

9. “Hoisting” of declarations to the beginning of functions is associated with which language?

   A. C B. JavaScript C. Scheme D. Pascal

10. PL/0 and Pascal-S are examples of which kind of semantics?

   A. attribute grammar B. denotational C. operational D. two-level grammar

Long Answer.

1. Give the result of each Scheme program. 20 points

   a. `(cons (car (cdr '(a (b (c d) e) f) )) (cdr (car '((a b c d e) (f g)))))`

   b.

```
(define (prob2 x)
  (cond
    ((eq? x 0) 1)
    (else
     (* x (prob2 (- x 1))))))
(prob2 5)
```

   c.

```
(define (prob3 lat)
  (define (help a l)
    (cond
      ((null? l) #f)
      ((eq? a (car l)) #t)
      (else (help a (cdr l)))))
  (cond
    ((null? lat) #f)
    (else
     (help (car lat) (cdr lat)))))
(prob3 '(x b c e f g a d h x))
(prob3 '(x b c e f g a d h))
(prob3 '())
(prob3 '(x))
```

2. Give grammar productions, railroad diagrams, or regular expressions for the following. 20 points

   a. Integers of arbitrary magnitude that are divisible by 5.

   b. A string using the symbols a, b, and c such that there are no b’s after the last (rightmost) c and there are no a’s after the last (rightmost) b.

   c. Pascal expressions over the constants 0 and 1, the binary operators *, +, and =, along with parentheses. Your answer should be suitable for developing a recursive-descent parser.

3. Write a Scheme function that takes a simple unordered list of integers and brings the smallest value to the front (e.g. head or car). If the smallest value appears several times, then the first occurence will be the one brought to the front. 20 points

```
> (smallToFront '(3 4 2 5 1))
'(1 3 4 2 5)
> (smallToFront '(1 3 4 2 5 1))
'(1 3 4 2 5 1)
```

```
> (smallToFront '(11 3 4 2 5 10))
'(2 11 3 4 5 10)
> (smallToFront '())
'()
> (smallToFront '(100))
'(100)
> (smallToFront '(11 3 2 4 2 5 2 10 2 2))
'(2 11 3 4 2 5 2 10 2 2)
> (smallToFront '(2 3 4 5 1 2 3 4 5 1 2 3 4 5 1))
'(1 2 3 4 5 2 3 4 5 1 2 3 4 5 1)
```

CSE 3302 Name ________________________________

Test 3

Fall 2013

Multiple Choice. Write your answer to the LEFT of each problem. 4 points each

1. Which language has immutable strings?
   A. C B. C++ C. JavaScript D. Pascal
2. `delete` is used in JavaScript to:
   A. invoke garbage collection B. remove a property from an object
   C. remove DOM elements from html D. remove global variables
3. A new property may be added to a JavaScript object using:
   A. assignment B. `new` C. `Object.create()` D. prototypal inheritance
4. Omitting the `new` on a call to an intended constructor will bind `this` to:
   A. an array of arguments B. the global object
   C. the last instance created by this constructor D. the prototype
5. Which of the following JavaScript objects does not have a `length`?
   A. functions B. numbers C. arrays 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. Arbitrary array subscript ranges are included in which language?
   A. C B. Scheme C. Java D. Pascal
8. Which of the following will be treated like `false` (In JavaScript)?
   A. `5` B. `1/2` C. `"else"` D. `""`
9. In JavaScript, what is the value of the expression `0 && 1 && 2 && 3`?
   A. `0` B. `false` C. `3` D. `true`
10. Duff's device involves which PL construct?
    A. C union B. C switch C. Java switch D. C varargs

Long Answer.

1. Give equivalent 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! (20 points)
   a. `result = a <= 10 && b > 13;`
   b. `result = c < 20 || d >= 17;`
   c. `result = !(e <= 25 || f > 55) || g <= 66;`
2. Code a JavaScript closure `appender` that will a) initialize a protected variable to `""` and b) on each call to `append` with a string as the argument will append that string (to the protected variable) and return the result. (20 points)

Example use:

```
c1=appender();
c2=appender();
c1.append("Michael ");
c2.append("Kobe ");
console.log(c1.append("Jordan"));
console.log(c2.append("Bryant"));
```

Result:

```
Michael Jordan
Kobe Bryant
```

3. Suppose a Pascal array is to be stored starting at location `20000` and is declared:
   `c: array[16..70,30..33,5..7] of integer;`
   If one `integer` takes two bytes, what is the location of `c[35,31,6]`? (20 points)

CSE 3302/5307 Name ________________________________

Test 4

Fall 2013

Multiple Choice. Write your answer to the LEFT of each problem. 5 points each

1. Keyword parameters give flexibility in:

   A. achieving overloading

   B. making call-by-name work effectively

   C. overriding the reserved words of a language

   D. the order in which parameters are passed

2. The difference between actual parameters and formal parameters is:

   A. actuals are in the caller, formals are in the called subprogram

   B. actuals are in the called subprogram, formals are in the caller

   C. actuals are call-by-value, formals are call-by-name

   D. no difference

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. PL/0 uses dynamic links to:

   A. Update the display table

   B. Return from a called procedure

   C. Place an integer on the stack

   D. Reference data

5. `(cons (car (cdr '(h (i j k) l m))) (cdr '(d e (g f) (a b c))))` will result in:

   A. `'((e (g f) (a b c)) i j k)`

   B. `'((i j k) e (g f) (a b c))`

   C. `'(((g f) (a b c)) i j k)`

   D. `'((i j) e (g f) (a b c))`

6. `(car (cons (car (cdr '(a (b c) (d e)))) (cdr '(((f g)) h i (j k)))))` will result in:

   A. `'(b c)`

   B. `'((b c))`

   C. `'(d e f g)`

   D. `'((b c) h i (j k))`

7. Continuations could be viewed as being a generalization of:

   A. activation records B. combinator C. exceptions D. functions

8. Call-by-name is associated with which language?

   A. ALGOL 60 B. C C. JavaScript D. Pascal

9. ## may be used in a C macro to:

   A. apply the macro in a call-by-name fashion

   B. concatenate tokens

   C. simulate Pascal `var` argument passing

   D. stringify tokens

10. The static link and dynamic link for an activation record will be the same when:

    A. never

    B. a procedure has called itself

    C. the call is from a nested procedure

    D. in all cases

What is the result of executing this Scheme code? 10 points

```
(define y 4)

((lambda (x y z)
   (x y z))
 (lambda (y z)
   ((lambda (x) (+ 8 z))
    5))
 7 11)
```

Long Answer. 20 points each.

1. Give Scheme code to compute the nesting depth of an S-expression, i.e. the empty list (or a single atom) has depth 0 and an S-expression with nested S-expressions with maximum depth *k* would itself have depth *k* + 1.

```
> (depth '1)
0
```

```
> (depth '())
0
> (depth '(1 2 3 4))
1
> (depth '(1 2 3 4 (1 2 3 4)))
2
> (depth '(d e (g f) (a b c)))
2
```

2. Give Scheme code for a predicate that returns `#t` if and only if its argument has some atom repeated twice, but not necessarily together. The argument will always be a list - no error checking is required. Efficiency is not an issue.

```
> (duplicate? '())
#f
> (duplicate? '(1 2 3 4))
#f
> (duplicate? '(1 2 3 4 (1 2 3 4)))
#t
> (duplicate? '(d e (g f) (a b c)))
#f
> (duplicate? '(d e (g f) (a b c d)))
#t
```