Name: _____   CS Username: _____

Grade: _____   Blackboard Username: _____

- Be sure to begin by printing your *name* and both *usernames* clearly in the spaces provided above.

- If you find a question ambigous: write the most reasonable assumptions you can think of near the question, and then answer the question under these assumptions.

- The exam is graded as 100 percentage points. The points available for individual questions are shown in parentheses.

1. Which language is considered the first high-level programming language? (mark **one**) (4 points)

   (a) Ada
   (b) Fortran
   (c) Lisp
   (d) Pascal

2. The main strength of Lisp is in the area of (mark **one**) (4 points)

   (a) Business computation
   (b) Symbolic computation
   (c) Scientific and numerical computation
   (d) System programming

3. What language is mainly intended for *business programming*? (mark **one**) (4 points)

   (a) Cobol
   (b) Basic
   (c) C++
   (d) Fortran 77

4. What language is mainly intended for *numerical programming*? (mark **one**) (4 points)

   (a) Basic
   (b) Fortran 77
   (c) Ada
   (d) Pascal

5. What language was the first *block structured* language? (mark **one**) (4 points)

   (a) C++
   (b) Fortran 77
   (c) Cobol
   (d) Algol 60

6. What language was the first to have the concept of a *class* for data abstraction? (mark **one**) (4 points)

   (a) C++
   (b) Algol 60
   (c) Simula 67
   (d) Smalltalk-80

7. Which language(s) is/are *object oriented* or hybrids with object oriented features? (mark **one or more**) (4 points)

   (a) Fortran 77

   (b) Ada 95

   (c) Smalltalk-80

   (d) Cobol

8. Which of the following language(s) is/are *functional languages*? (mark **one or more**) (4 points)

   (a) PL/I

   (b) Lisp

   (c) Haskell

   (d) Cobol

9. Which of the following languages are strongly typed (i.e. type errors are always detected)? (mark **one or more**) (4 points)

   (a) C++

   (b) Java

   (c) Ada

   (d) Pascal

10. Which of the following classes of programming languages is imperative? (mark **one**) (4 points)

    (a) Dataflow

    (b) Functional

    (c) Logical

    (d) Procedural

11. What is a fixed format language? (mark **one**, 4 points)

    (a) A language with a fixed number of keywords

    (b) A language in which the relative positions of tokens with respect to eachother is important rather than the position of the tokens on the page

    (c) A language in which indentation is significant, hence the amount of spacing is meaningfull and influences the execution of a program

    (d) A language with formatted read and write constructs

12. In a *strongly typed* language . . . (mark **one**) (4 points)

    (a) . . . all objects are polymorphic

    (b) . . . type errors are always detected

    (c) . . . recursion is not supported

    (d) . . . all variables are statically allocated

13. Some languages perform array bounds checking, which means that array subscript values (indices) must stay within the specified array bounds. The checks cannot always be performed at compile time, because array subscript values often depend on program input data. What kind of error occurs when an array subscript value is out of bounds at run time? (mark **one**) (4 points)

    (a) Lexical error

    (b) Syntax error

    (c) Static semantic error

    (d) Dynamic semantic error

14. Which of the programming tools below is also called a *virtual machine*? (mark **one**) (4 points)

   (a) Preprocessor

   (b) Compiler

   (c) Interpreter

   (d) Linker

15. What is the value of the Scheme expression `(cons (car (cdr (cdr '(1 2 3 4 5)))) '(6 7))` ? (mark **one**) (4 points)

   (a) `(1 2 3 6 7)`

   (b) `(1 2 3)`

   (c) `(3 4 5)`

   (d) `(3 6 7)`

16. Consider the Scheme function `f` defined below:

```
(define f
  (lambda (n k)
    (cond
      ((> n k) (- n k))
      ((= n k) 0)
      (else    (- k n))
    )
  )
)
```

   What is the resulting value of `(f 3 5)` in Scheme? (4 points)

   (a) `0`

   (b) `1`

   (c) `2`

   (d) `3`

17. A compiler can be divided into a "front end" and a "back end".

   (a) Name three steps in (or components of) the front end compiler process. (6 points)

   (b) Name *and describe* a convenient intermediate form that describes the result of applying the front end compilation process. (4 points)

18. Consider the following attributed grammar:

```
<p2> -> '['<p1>']'    p2.count = p1.count + 2
<p>  -> e             p.count = 0
```

(a) Describe in words the collection of syntactically correct strings satisfying this grammar. (4 points)

(b) Describe in words the semantics determined by the grammar attributes. (4 points)

(c) Draw a decorated parse tree for the input "[[]]": (8 points)

19. Draw the layout of a *subroutine frame*. Identify the fields and describe their purpose. (10 points)