CSCI 330 Spring 1999
MIDTERM EXAM
Thursday 99/2/18—Two Pages

1 Short Questions–1 point each

1. What programming language has dominated scientific computing over the **Answer:** FORTRAN past 35 years?
2. What programming language has dominated business applications over the past 35 years? **Answer:** COBOL
3. What programming language has dominated artificial intelligence over the past 35 years? **Answer:** LISP (Also acceptable: LISP and Prolog)
4. In what language is UNIX written? **Answer:** C
5. How can user-defined overloading harm the readability of a program? **Answer:** A reader must refer to the declarations in which an operator is overloaded to be certain of the meaning of the operator when the operator is used.
6. Fill in the correct word in place of *mystery* in the following sentence: The Ariane 5 launcher disaster occurred because of an uncaught *mystery* during a type conversion. **Answer:** Exception
7. What is the name of the category of programming languages whose structure is dictated by the von Neumann computer architecture? **Answer:** Imperative
8. What are the three fundamental features of an object-oriented programming language? **Answer:** Encapsulation, inheritance, and dynamic type binding (or polymorphism)
9. Who developed LISP? **Answer:** John McCarthy (at MIT)
10. In which language was block structure introduced? **Answer:** ALGOL 60
11. What is a nonprocedural language? **Answer:** One that allows specification of what has to be computed rather than just how a computation is to be carried out.
12. What language used orthogonality as a major design criterion? **Answer**: ALGOL 68

13. Why does C++ include the features of C that are known to be unsafe? **Answer:** Backwards compatibility

14. What are two reasons that Java is safer than C? **Answer:** Here are three reasons: Implicit storage deallocation for heap-allocated objects (garbage collection), no pointers (and therefore no pointer arithmetic), and no implicit type coercions.

15. What is the result of applying `!+` to `<1 2 3>` in FP? **Answer:** 6

2 ML–35 points

1. What is the type of x in the ML expression $x + 2$? (1 point) **Answer:** int

2. In ML, you are not required to indicate the type of all variables used in a program. However, ML is considered a strongly typed language. Explain this apparent contradiction. (4 points) **Answer:** ML uses a sophisticated type inference algorithm, so that every variable's type is determined from sufficient clues.

3. Write a function *genOnes(n)* that constructs a list of n ones. For example, *genOnes(2)* produces `[1,1]`. (11 points) **Answer:** `fun genOnes 0 = [] | genOnes n = 1 :: genOnes(n-1);` *Alternative Answers* use append instead of cons, `if then else` instead of patters.

4. Write a function *genBinInt(n)* that produces a binary representation of the integer $2^n - 1$ as a pair $(n, [1, 1, \ldots, 1])$, where there are n ones in the list. For example, *genBinInt(2)* produces `(2, [1,1])`. **Hint:** Use genOnes. (4 points) **Answer:** `fun genBinInt n = (n, genOnes n);`

5. Write a function *myhead* that takes a non-empty list as argument, and returns the list without its last element. Use patterns and cons. **Hint:** Use two patterns: one for a list of only one element, one for a list with more than one elements (15 points). **Answer:** `fun myhead (x::[]) = [] | myhead (x::xs) = x :: myhead(xs);` *Alternative Answer :* `fun myhead (x::[]) = [] | myhead(x::xs) = [x]@myhead(xs);` (The version with cons is preferable, because it is more efficient; variants that involve use of `if then else` instead of patterns are penalized; variants that use reverse (without cons) are also penalized.)