

CS 4100, Spring 2013 – Programming Languages
Final Exam
Due May 24, 2013

- This is a take-home exam. All answers must be your own work. You may:
 - Discuss this exam with me,
 - Use our text book, class notes and handouts, books about Lisp,
 - **No other sources are allowed.**
 - **Any outside sources must be cited.**
- To receive full credit, **show your work** and **write legibly**.
- If you need clarification about any of the problems, please ask me.

Name: _____

For the problems on this exam you may talk to me, consult your textbook, course materials, books about Lisp, but **NOT** other students. Any outside consultations/sources must be cited.

1. Associate each of the following concepts with one of the languages we covered (pseudo code, FORTRAN, Algol 60, Pascal, or Lisp). If the concept is closely associated with more than one language, choose the one that first introduced it in an important way. Explain your answer to each concept in the space to the right by defining/explaining the concept and providing some perspective about its value and use.

_____ Call by name

_____ Call by value-result

_____ Computed GOTOs

_____ Dynamic Scoping

_____ For loop

_____ Garbage Collection

_____ Recursion

_____ Statement Labels

_____ Strong Typing

_____ User-defined Enumerated Types

2. A new language, Boa, with a block structured syntax similar to Pascal uses either pass-by-reference or pass-by-value as a parameter passing mechanism, but you don't know which. Design a test program you can use to find out and explain how the results of your program will enable you to detect the parameter passing mechanism.

3. Assume (because it is not strictly true in the real world) that an email address consists of a non-empty string of letters and digits, followed by an “at” sign (@), followed by a machine name made up of a series of one or more non-empty strings of letters and numbers separated by periods (.) and terminated by either “.com” or “.edu”. Write a BNF (**not** extended BNF) description of such an email address.

Examples:

myself@cs.csustan.edu

2million2@hotmail.com

1579@17.25.20.3.com

2bnot2b@hamlet.eggs.spam.edu

4. Write a recursive function `count` in LISP to count the number of atoms in a list no matter what their level of nesting. You may assume that `(atom x)` tests whether `x` is an atom and returns either true or false depending on the results of the test. For example:

```
(count '(a b (c 4) ((99)) nil t))
```

7

5. Consider the following segment of code:

```
Procedure M
var x: integer;

    Procedure A
    var a, b, c: integer;

        Procedure B (x: integer)
        var b, e: integer;
        Begin {B}
            ...
            a := b + c - e;
            ...
        End B;

    Begin {A}
        ...
        B(7)
        ...
    End A;

Begin {M}
    ...
    A()
    ...
End M;
```

Assuming the definition of M is at static nesting level 0. Draw the runtime stack that will be in effect at the execution of “a := b + c - e;”. Show the static and dynamic links, and the current positions of the EP and SP pointers.

What is the static distance to c’s environment of definition and the offset within the environment when “a := b + c - e;” is executed?