

CS 4100, Fall 2019 – Programming Languages

Final Exam

Due December 13, 2019 by 1:15 pm

- 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,
  - **No other sources are allowed.**
- To receive full credit, **show your work** and **write legibly**.
- If you need clarification about any of the problems, please ask me.

Name: \_\_\_\_\_

1. Associate each of the following concepts with one of the languages we covered (pseudo code, FORTRAN, Algol 60, Pascal, or Lisp) Write the language in the blank to the left. 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.

\_\_\_\_\_ Pass by reference

\_\_\_\_\_ Pass by Value-Result

\_\_\_\_\_ Assigned GOTOs

\_\_\_\_\_ Dynamic Scoping

\_\_\_\_\_ For loop

\_\_\_\_\_ Garbage Collection

\_\_\_\_\_ Recursion

\_\_\_\_\_ Sets

2. A new language, Houston, with a block structured syntax similar to Algol 60 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 which parameter passing method it uses (use and Algol-like pseudocode). Then, explain clearly 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. This questions comes from Chapter 6, which you will need to read on your own. Consider the following segment of code:

```
Procedure M
var x: integer;

    Procedure A
    var a, b, c: integer;

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

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

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

Assuming the definition of Procedure M is at static nesting level 0. Draw the runtime stack that will be in effect at the execution of “a := b + c - d;”. 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 - d;” is executed?