

# Compilers [Fall 2017]

## Test III

NAME: \_\_\_\_\_

### Instructions:

- 1) This test is 9 pages in length.
- 2) You have 2 hours to complete and turn in this test.
- 3) Short-answer responses and essays will be graded on how clearly you've communicated the necessary ideas. Write in complete English sentences. Avoid bulleted lists.
- 4) Always show enough work to convince me that you know what you're doing (i.e., you're not just guessing).
- 5) Always make the assumptions we made in class, regarding code generation in dj2dism.
- 6) This test is closed books, notes, papers, friends, neighbors, phones, smartwatches, etc.
  
- 7) Write and sign the following: "I pledge my Honor that I have not cheated, and will not cheat, on this test."

\_\_\_\_\_  
\_\_\_\_\_

Signed: \_\_\_\_\_

1. [8 points]

Prove that for all finite languages  $L$  there exists a DFA  $D$  such that  $D$  recognizes  $L$ .

2. [8 points]

Explain how generational garbage collectors work, at the level of detail discussed in class.

3. [18 points]

a) Define a grammar in  $SLR \cap LL(1) \setminus LR(0)$ , or explain why no such grammar exists.

b) Define a grammar in  $SLR \cap LR(0) \setminus LL(1)$ , or explain why no such grammar exists.

c) Define a grammar in  $SLR \cap LL(1) \setminus LALR$ , or explain why no such grammar exists.

d) Define a grammar in  $LL(1) \cap LALR \setminus LR(0)$ , or explain why no such grammar exists.

4. [15 points]

What is the maximum number of temporaries that could ever be on the stack when running an error-free program generated by dj2dism? If such an upper bound exists, show an example source program that causes dj2dism's output to exhibit this upper bound. On the other hand, if no such upper bound exists, explain why.

5. [20 points]

Show example dj2dism output for the following input. `main { nat n; n=1; printNat(n); }`

6. [20 points]

Although program-wide vtables are standard in OOPLs, as discussed in class, let's be skeptical. What would code-generation of an expression like  $e.ID(arg_1 \dots arg_n)$  look like without jumping to a vtable? Is this alternative a good idea? Why or why not?

7. [11 points]

List the 8 code optimizations we discussed in class, and for each one, illustrate the optimization on a small example. One of your examples should include spilling.



**Undergraduates stop here. The remaining problem is for graduate students.**

8. [5 points]

Define an algorithm to remove right recursion from a given grammar.