

# Quiz 3

EECS665 - Compiler Construction  
2019, Fall

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

Student ID: \_\_\_\_\_

## DO NOT OPEN UNTIL INSTRUCTED!

Before the Quiz starts:

- Read all of the instructions on this page
- Write your name and student ID on this page
- Retrieve your page of notes and writing materials
- Put all other materials away and silence your devices

After the Quiz starts:

- Write your student ID (**not** your name) on all subsequent pages
- If you feel a question is wrong or impossible, notify course staff.
- Announcements / corrections will appear on the projector
- Turn in all your related paper when finished, including:
  - your notes page
  - the provided quiz itself
  - provided reference pages
  - provided scratch paper
- You may leave when done (no new material will be presented).
- Work quickly, move on if you are stuck.

Feel free to draw something **not spooky**  
in the box below to pass the time

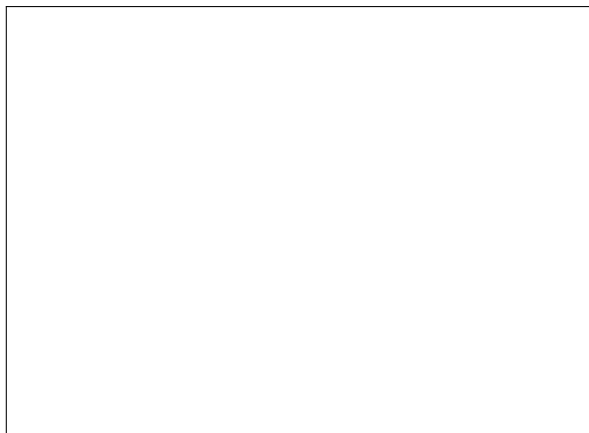
Total Questions: 5

Time Limit: 35 minutes

Total Pages:

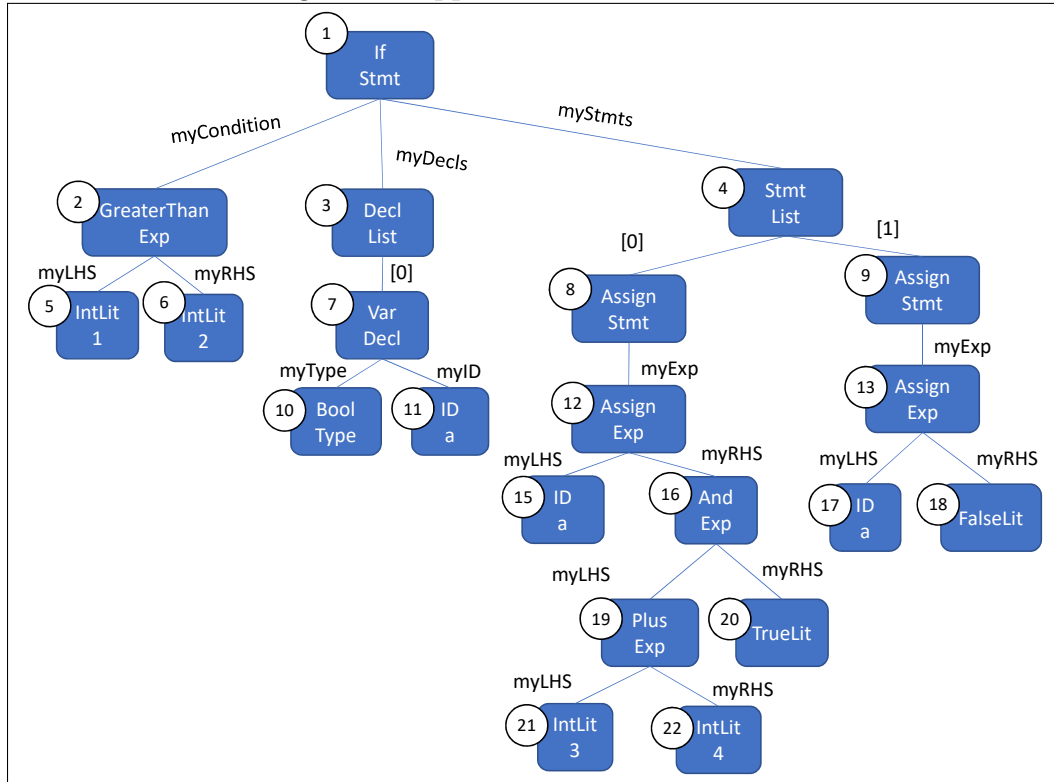
- 6 pages total

Score: \_\_\_\_\_ / 50 pts



**QUESTION 1 (10 POINTS)**

Consider the following AST snippet:



Indicate, at each node, what type would be assigned to that node. Also, indicate if type checking would report an error at that node.

- |                 |             |                 |             |
|-----------------|-------------|-----------------|-------------|
| 1. type: _____  | report: y/n | 12. type: _____ | report: y/n |
| 2. type: _____  | report: y/n | 13. type: _____ | report: y/n |
| 3. type: _____  | report: y/n | 14. type: _____ | report: y/n |
| 4. type: _____  | report: y/n | 15. type: _____ | report: y/n |
| 5. type: _____  | report: y/n | 16. type: _____ | report: y/n |
| 6. type: _____  | report: y/n | 17. type: _____ | report: y/n |
| 7. type: _____  | report: y/n | 18. type: _____ | report: y/n |
| 8. type: _____  | report: y/n | 19. type: _____ | report: y/n |
| 9. type: _____  | report: y/n | 20. type: _____ | report: y/n |
| 10. type: _____ | report: y/n | 21. type: _____ | report: y/n |
| 11. type: _____ | report: y/n | 22. type: _____ | report: y/n |

## QUESTION 2 (15 POINTS)

Student ID: \_\_\_\_\_

### PART I (10 POINTS)

Write out the source code snippet that could have induced the following 3AC program snippet. Assume that no variable is used uninitialized. Make sure any variables used in the below snippet are declared in your source code snippet.

```
L1:  enter hominem
L2:  getin 1 one
L3:  getin 2 two
L4:  tmp := one + 3
L5:  tmp2 := two / 4
L6:  tmp3 := one - two
L7:  setout tmp3
L8:  leave
L9:  enter te
L10: iffalse three goto L14
L11: three := 1
L12: goto L14
L13: three := 2
L14: nop
L15: call memento
L16: leave
```

### PART II (5 POINTS)

Draw the control flow graph for the function(s) in the above 3AC snippet. You may use the labels to indicate which 3AC quads go in which basic blocks.

### QUESTION 3 (5 POINTS)

Student ID: \_\_\_\_\_

Explain the difference between static typing and dynamic typing

Give the name of two different dynamic typing schemes

What is type safety? Give an example of a language that is not type safe.

## QUESTION 4 (10 POINTS)

Student ID: \_\_\_\_\_

Show a program and describe two optimizations (A and B) such that optimization B can be applied only after optimization A has been performed. You may not use constant folding for either optimization A or B.

## QUESTION 5 (10 POINTS)

Student ID: \_\_\_\_\_

The following CFG has each basic block annotated with its IN set computed after a flow-based dataflow analysis for constant propagation. Write out the 3AC that might fit in each of the basic blocks P1 through P4 to fulfill these IN sets. If no 3AC is possible, write INVALID for that block. Note that the block may be composed of multiple 3AC instructions/quads.

