

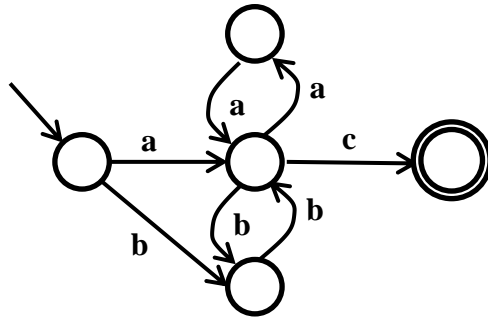
SFU CMPT 379 - Compilers
Summer 2021
Final

150 minutes. 200 points. According to my calculations, that's four points every 3 minutes.
I will be available on zoom (same link as office hours) during the exam to answer questions.
Open book, open notes. Internet may be used only to view lecture slides, format answers, or ask me questions. You are given 25 minutes after the exam period to format and upload your answers to Canvas; this time is not to be used for answering questions. By submitting answers, you are certifying that you have adhered to all exam procedures and academic honesty guidelines.

There are 10 questions. Manage your time carefully.

1. (6 points; 3 each)
 - a. What is your name?
 - b. What is your student number?
2. (24 points; 4 each) State which phase of a typical C++ or java compiler normally performs the following tasks. If something is done at run-time, then it's the code generator's responsibility.
 - a. Creating the control-flow graph for a procedure.
 - b. Ensuring that every opening-brace '{' has a corresponding closing-brace '}'.
 - c. Determining the type of an expression.
 - d. Eliminating useless code.
 - e. Ensuring that every "break" statement is inside a loop.
 - f. Recognizing a floating-point constant.
3. (16 points) What is the difference between a *primitive* type and a *compound* type?
Describe these in relation to the programmer, the language, and the computer on which the program is run. Give examples from java or C++.
4. (16 points) Using the following grammar, give a leftmost derivation for $c*i*i + c$
$$\begin{aligned} E &\rightarrow T \mid T+E \\ T &\rightarrow F \mid F * T \\ F &\rightarrow i \mid c \end{aligned}$$
5. (16 points) Draw a DFA (not NFA) transition diagram for the regular expression $(ab|ba)^+ba$

6. (16 points) Write a single compact regular expression for the language accepted by the following DFA.



7. (45 points total)

a. (15 points) Rewrite the following three-address code in SSA form.

```

1.      a = 2 * 4
2.      b = 2 * a
3.  loop: t1 = a + b
4.      t2 = a * a
5.      branch t2 < t1 exit
6.      a = b + c
7.      b = 2 * b
8.      goto loop
9.  exit: b = b - 1
10.     a = a - 1
  
```

b. (10 points) Draw the CFG of the SSA form, with the code inside the blocks of the CFG.

c. (10 points) What are the LIVE_OUT variables of each block of the CFG from (b)?

d. (10 points) What are the upwards-exposed variables of each block of the CFG from (b)? (limit your search to each basic block; not a full dataflow)

8. (20 points) Suppose you were to have to implement global and static variables in your Bilby compiler. Global variables have their definitions/initializations intermixed with procedure definitions before **main**. Static variables are variables in a procedure that are essentially global variables but they are only accessible in the procedure they are defined and initialized in. Give the details as to how this could be done, and what the issues with these features are. Do talk about what is to be done in each phase, but do not give code.

9. (20 points) Suppose you were to have to implement separate compilation for Bilby. Bilby would allow a program to appear over several files, only one of which has a main block. The others contain function definitions only. How do you get them each to produce an output (ASM?) file, which can be easily combined together to give a working ASM program? Give the details as to how this could be done, and what the issues with these features are. Do talk about what is to be done in each phase, and how the linking of the separate files would be done, but do not give code.

10. (21 points; 7 each) In the following CFG,
- which nodes dominate B_7 ?
 - which node is B_8 's immediate dominator?
 - which nodes postdominate B_3 ?

