

CSC465/2104 Test 1 2024 October 17 4:10pm

1 page, 3 questions, 42 marks, 50 minutes
Aids allowed: one letter-sized page, both sides
and the laws from the textbook, 14 pages

The value of each question is indicated in square brackets.

A blank answer is worth about one-third of the marks;
to that, marks will be added for readable and relevant and correct information,
and marks will be subtracted for unreadable or irrelevant or incorrect information.

- 1[9] Let a and b be integer variables. Express
 $a := a+b+1$. $b := a-b-1$. $a := a-b-1$
as simply as possible without using quantifiers, assignments, or sequential compositions.
Proof required.
- 2 We have a two-tailed **if** programming notation with the syntax
if binary expression **then** program **else** program **fi**
Most programming languages also have a one-tailed **if** with a syntax like
if binary expression **then** program **fi**
It is executed by first evaluating the binary expression, and then executing the program if
and only if the binary expression's value was \top .
- (a)[3] Define the one-tailed **if** formally.
- (b)[12] Let n be a natural variable, and let t be time measured recursively. Define
 $countdown = n \geq 0 \Rightarrow t' = t + n$
Prove the refinement
 $countdown \Leftarrow \mathbf{if } n > 0 \mathbf{ then } n := n - 1. t := t + 1. \mathbf{fi}$
- 3[18] Let n be natural and let s be a natural variable. Write a program for $s' = n^2$ without
using multiplication or exponentiation. Use a **for**-loop with **for**-index k and invariant
 $s = k^2$. Proof required.