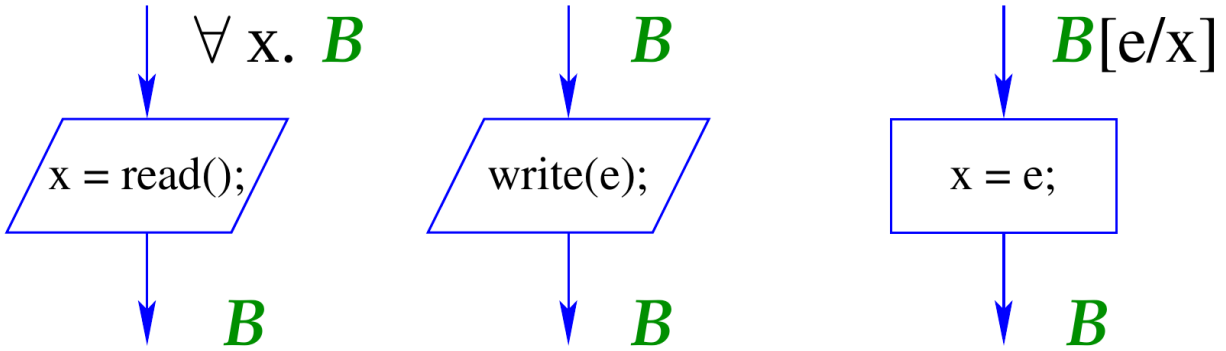
Week 3: Loop Invariants

Week 03 Tutorial 01 — MiniJava 2.0

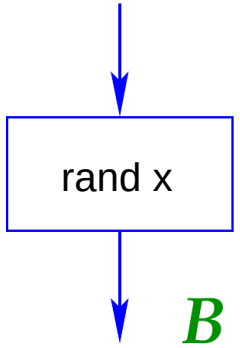


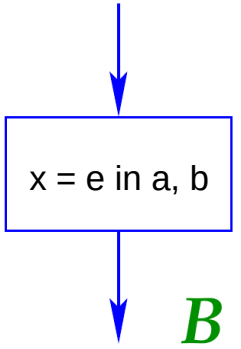
Week 03 Tutorial 01 — MiniJava 2.0

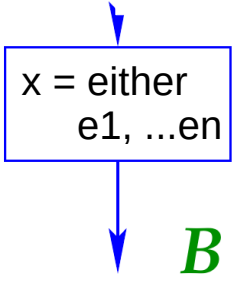
1. `rand x`:
   Assigns a random value to variable x,
2. `x = either` $e_0, \ldots, e_k$:
   Assigns one of the values of the expressions $e_0, \ldots, e_k$ to variable `x` non-deterministically,
3. `x = e in a, b`:
   Assigns the value 1 to variable `x`, if the value of expression `e` is in the range $[a, b]$ and 0 if `e` is not in the range or the range is empty $(a > b)$,
4. `stop`:
   Immediately stops the program.

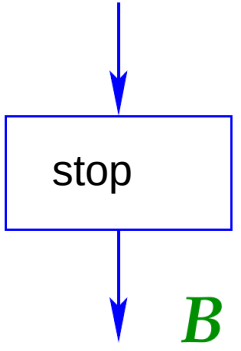
Define the weakest precondition operator $\mathbf{WP}[\![\ldots]\!](B)$ for each of these statements.

Week 03 Tutorial 01 — MiniJava 2.0

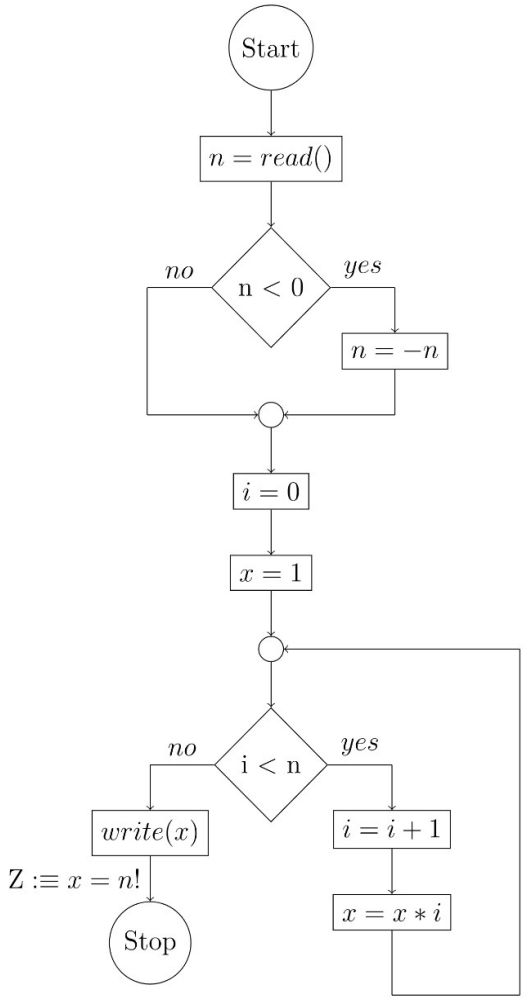








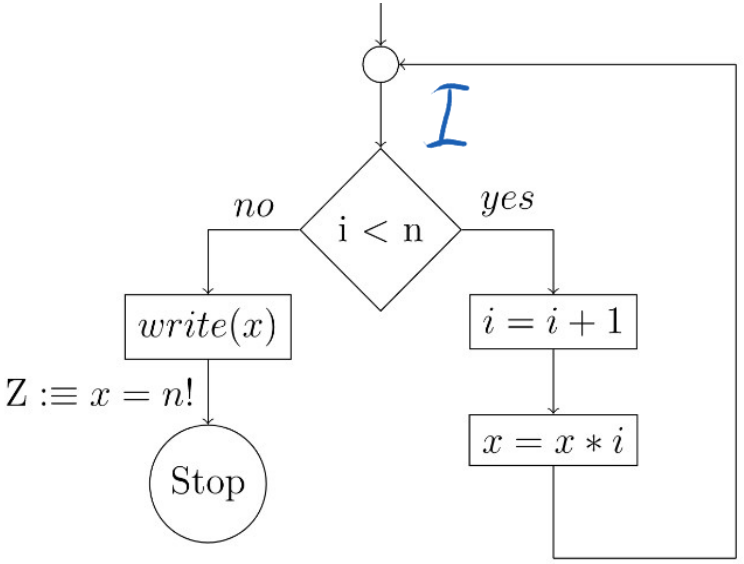
Week 03 Tutorial 02 — Loop Invariants



1. Discuss the problem that arises when computing weakest preconditions to prove Z.
2. How can you use weakest preconditions to prove Z anyway?

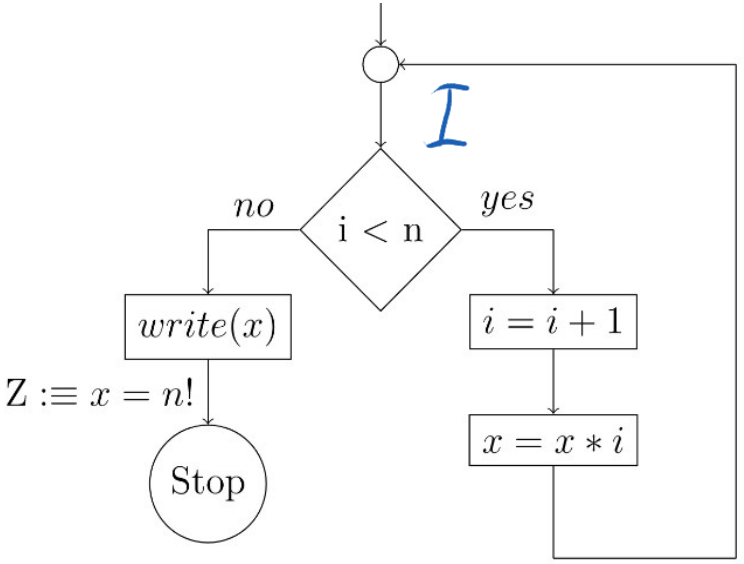
Week 03 Tutorial 02 — Loop Invariants

$I :\equiv x \geq 0$



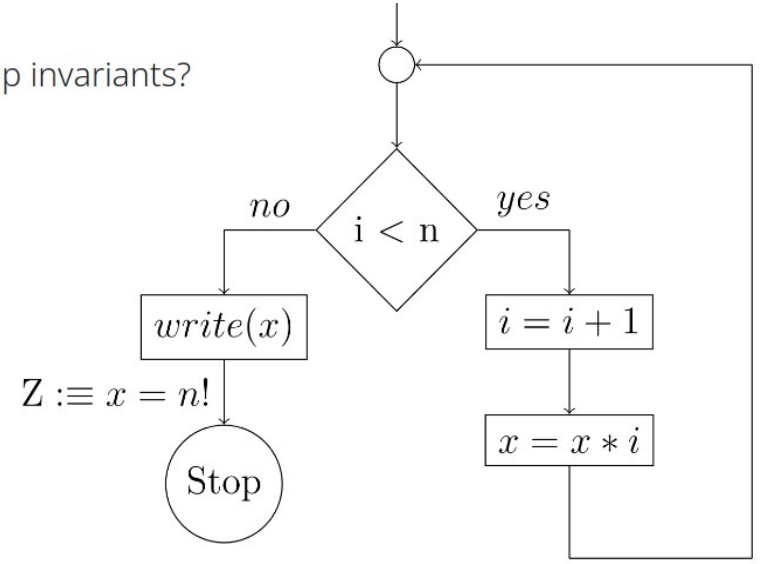
Week 03 Tutorial 02 — Loop Invariants

$I :\equiv i = 0 \wedge x = 1 \wedge n = 0$

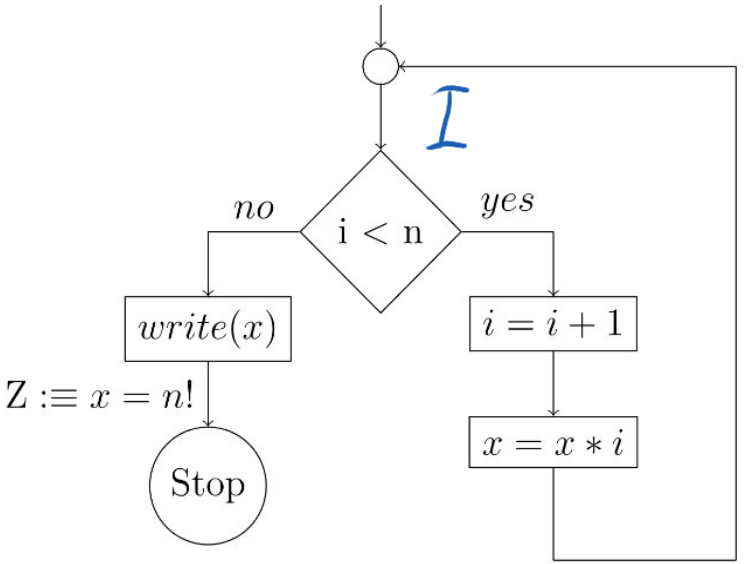


Week 03 Tutorial 02 — Loop Invariants

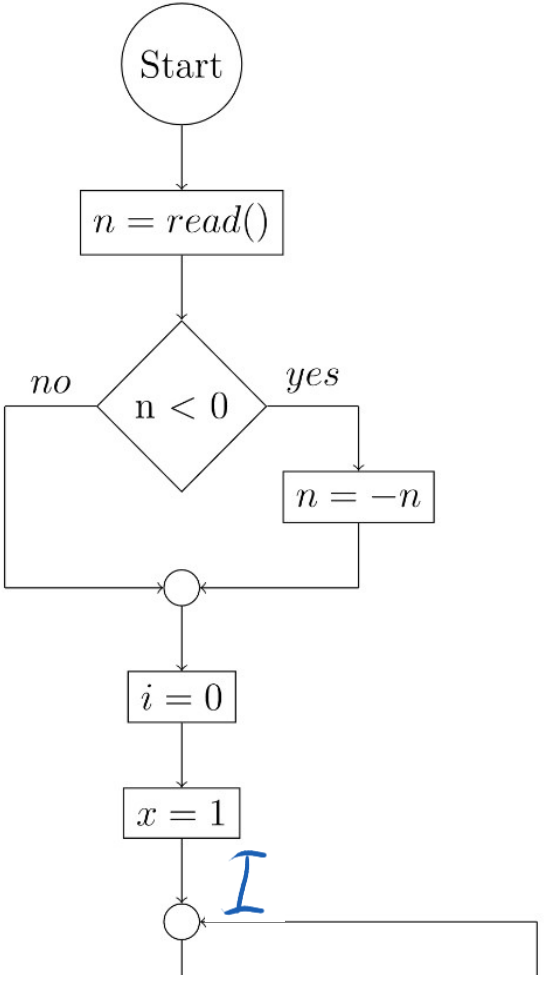
- a) How has a useful loop invariant be related to Z?
- b) What happens if the loop invariant is chosen too strong?
- c) What happens if the loop invariant is chosen too weak?
- d) Can you give a meaningful lower and upper bound for useful loop invariants?



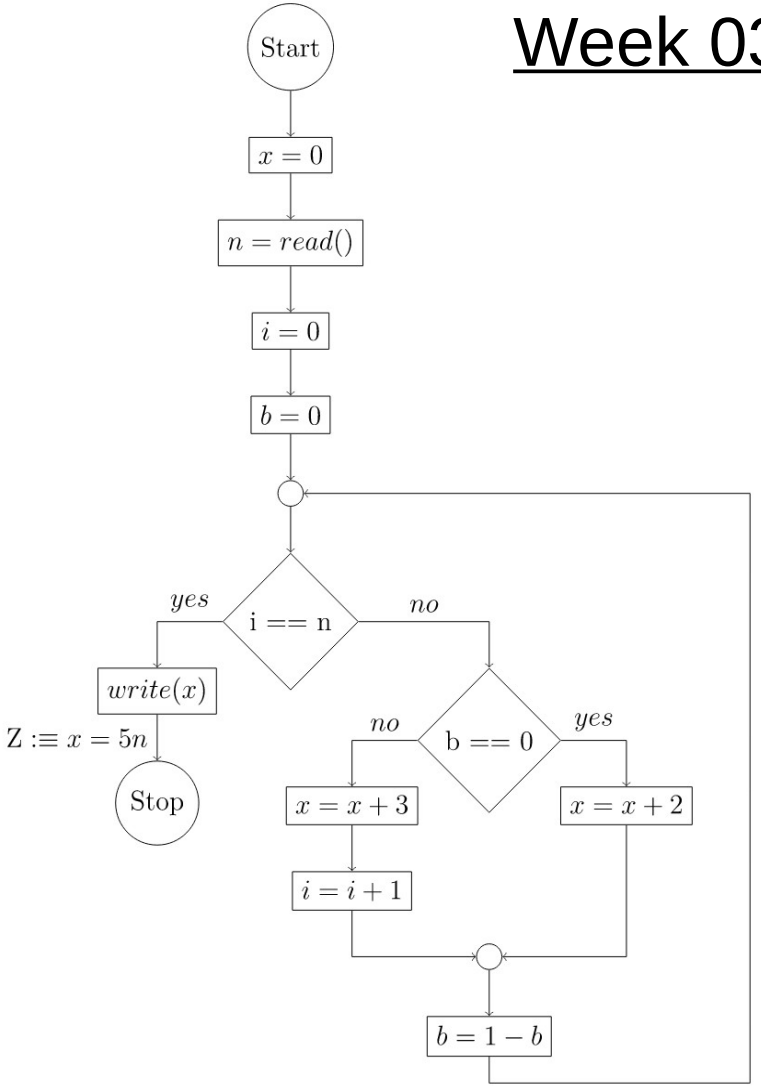
Week 03 Tutorial 02 — Loop Invariants



Week 03 Tutorial 02 — Loop Invariants



Week 03 Tutorial 03 — Two b, or Not Two b



Week 03 Tutorial 03 — Two b, or Not Two b