Functional Programming and Verification

Prof. Dr. H. Seidl, N. Hartmann, R. Vogler

WS 2018/19

Exercise Sheet 4

Deadline: 18.11.2018

General Information

Detailed information about the lecture, tutorials and homework assignments can be found on the lecture website[1]. Solutions have to be submitted to Moodle[2]. Make sure your uploaded documents are readable. Blurred images will be rejected. Use Piazza[3] to ask questions and discuss with your fellow students.

Simplifications

Make sure you simplify terms whenever possible. Overly complex proofs with huge formulas due to lack of any meaningful simplification will not be awarded with full points.

OCaml Setup

We are going to start OCaml programming in next week's exercises. Please prepare your machines accordingly and bring them to the sessions. A detailed installation guide will be published soon. Please check the website and Moodle.

Assignment 4.1 (L) Termination

In the lecture, you have learned how to prove termination of a MiniJava program. Discuss these questions:

1. How can you decide whether a termination proof is required at all?
2. What is the basic idea of the termination proof?
3. How has the program to be modified?
4. What has to be proven?
5. How is the loop invariant influenced?

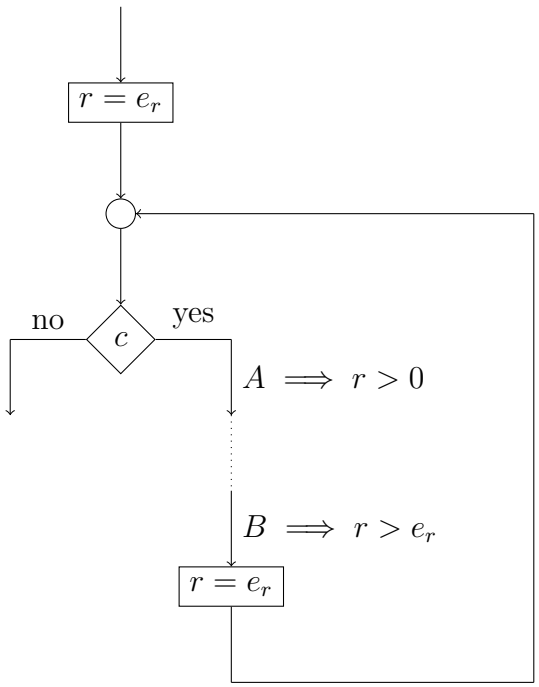
Suggested Solution 4.1

Consider this general structure of a loop:

[1] `https://www.in.tum.de/i02/lehre/wintersemester-1819/vorlesungen/functional-programming-and-verification/`

[2] `https://www.moodle.tum.de/course/view.php?id=44932`

[3] `https://piazza.com/tum.de/fall2018/in0003/home`



1. A loop-free program always terminates. If the program contains one or more loops, every single loop has to be considered. Classical `for`-style loops always terminate. Every other loop has to be checked.

2. A loop terminates if and only if the number of iterations is finite. Choose a variable of the program that is strictly increasing (or decreasing) in every single iteration. Then, if we can prove that there is an upper (or lower) bound the variable cannot reach, the loop must run only a finite number of iterations, because otherwise it would eventually reach the bound. The common strategy is to choose a variable that is decreasing by exactly 1 in every iteration and a lower bound of 0, although the proof might work with other values as well.

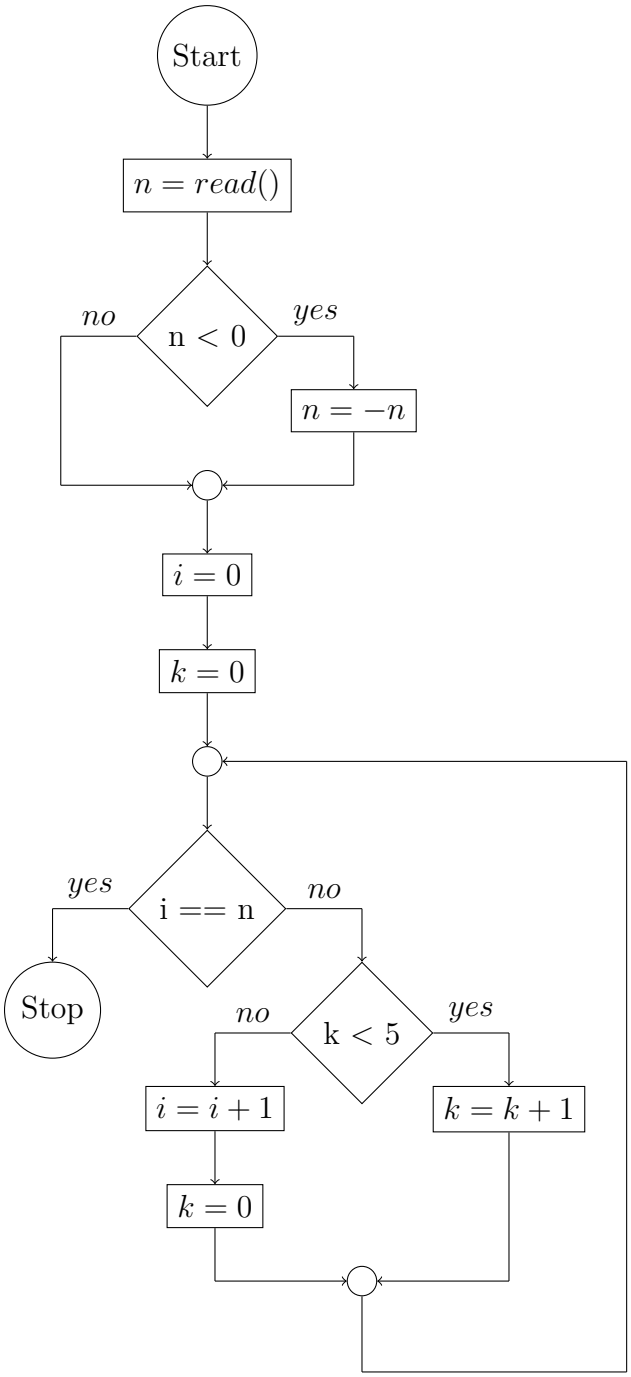
3. If the program does not provide a variable that satisfies these requirements, we introduce a new variable (typically named r) and compute it in such a way that it models this behavior before and after every iteration of the loop.

4. Now, we need to find locally consistent annotations in the program such that the assertion A at the beginning of the loop body enforces the lower bound ($A \implies r > 0$) and the assertion B at the end of the loop body (before the recomputation of r) guarantees that r has been decreased in every iteration ($r > e_r$, where e_r is the expression used to compute r).

5. The loop invariant typically contains $r = e_r$ as well as relations between the variables used in e_r.

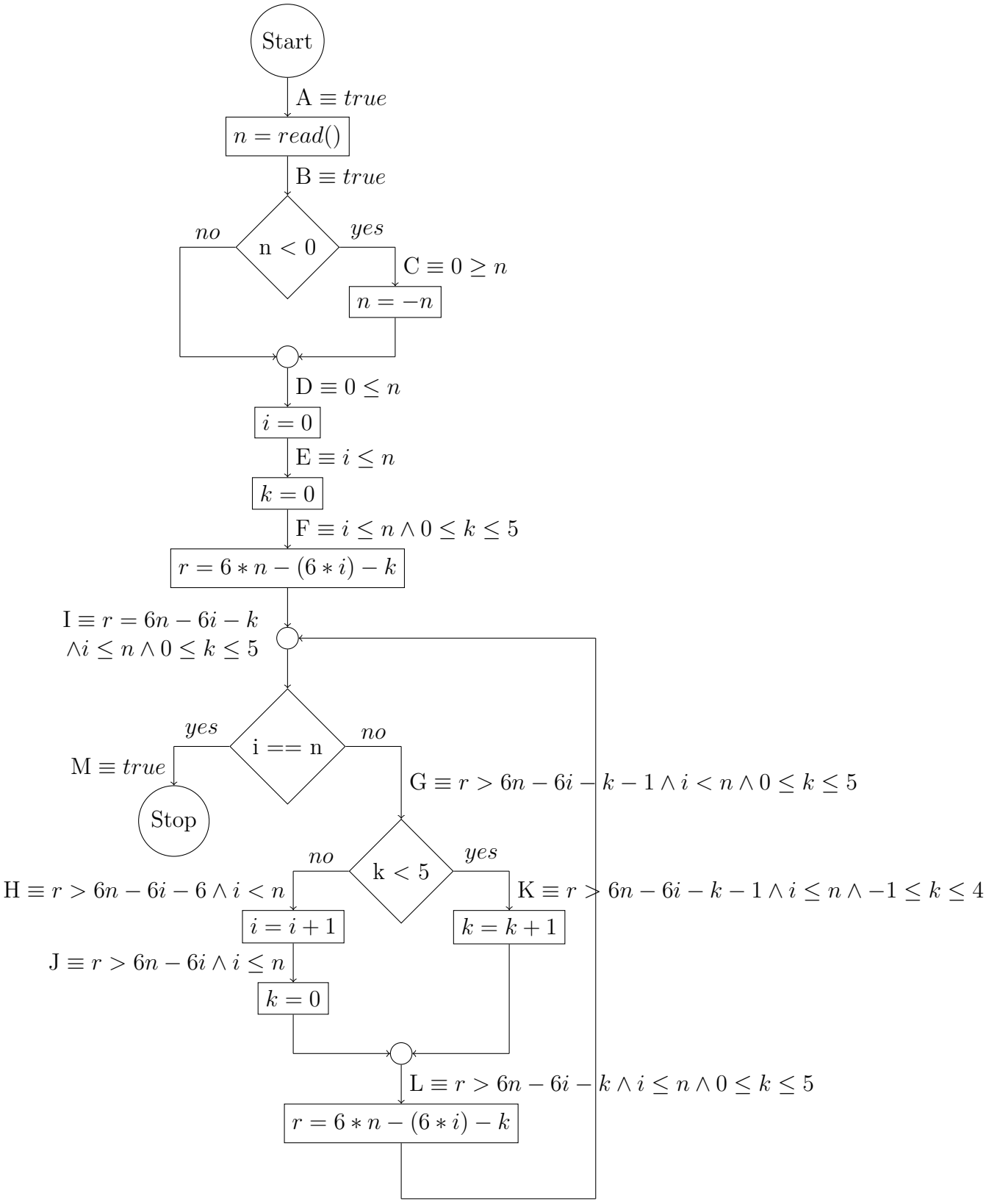
Assignment 4.2 (L) Counter Time

Prove that the following program does indeed terminate for all inputs.



Suggested Solution 4.2

We add an auxiliary variable $r = 6n - 6i - k$ to the control flow graph:



Now we have to find locally consistent annotations such that

- $G \implies r > 0$ and
- $L \implies r > 6n - 6i - k$

A suitable loop invariant is $I :\equiv r = 6n - 6i - k \wedge i \leq n \wedge 0 \leq k \leq 5$.

$$
\begin{aligned}
& \mathsf{WP}[\![\texttt{r = 6n - 6i - k}]\!](I) \\
\equiv\ & \mathsf{WP}[\![\texttt{r = 6n - 6i - k}]\!](r = 6n - 6i - k \wedge i \leq n \wedge 0 \leq k \leq 5) \\
\equiv\ & i \leq n \wedge 0 \leq k \leq 5 \\
\Longleftarrow\ & r > 6n - 6i - k \wedge i \leq n \wedge 0 \leq k \leq 5 \quad \equiv: L
\end{aligned}
$$

$$
\begin{aligned}
& \mathsf{WP}[\![\texttt{k = 0}]\!](L) \\
\equiv\ & \mathsf{WP}[\![\texttt{k = 0}]\!](r > 6n - 6i - k \wedge i \leq n \wedge 0 \leq k \leq 5) \\
\equiv\ & r > 6n - 6i \wedge i \leq n \quad \equiv: J
\end{aligned}
$$

$$
\begin{aligned}
& \mathsf{WP}[\![\texttt{i = i + 1}]\!](J) \\
\equiv\ & \mathsf{WP}[\![\texttt{i = i + 1}]\!](r > 6n - 6i \wedge i \leq n) \\
\equiv\ & r > 6n - 6i - 6 \wedge i < n \quad \equiv: H
\end{aligned}
$$

$$
\begin{aligned}
& \mathsf{WP}[\![\texttt{k = k + 1}]\!](L) \\
\equiv\ & \mathsf{WP}[\![\texttt{k = k + 1}]\!](r > 6n - 6i - k \wedge i \leq n \wedge 0 \leq k \leq 5) \\
\equiv\ & r > 6n - 6i - k - 1 \wedge i \leq n \wedge -1 \leq k \leq 4 \quad \equiv: K
\end{aligned}
$$

$$
\begin{aligned}
& \mathsf{WP}[\![\texttt{k < 5}]\!](H, K) \\
\equiv\ & \mathsf{WP}[\![\texttt{k < 5}]\!](r > 6n - 6i - 6 \wedge i < n, r > 6n - 6i - k - 1 \wedge i \leq n \wedge -1 \leq k \leq 4) \\
\equiv\ & (k \geq 5 \wedge r > 6n - 6i - 6 \wedge i < n) \vee (r > 6n - 6i - k - 1 \wedge i \leq n \wedge -1 \leq k \leq 4) \\
\Longleftarrow\ & (k = 5 \wedge r > 6n - 6i - k - 1 \wedge i < n) \vee (r > 6n - 6i - k - 1 \wedge i < n \wedge 0 \leq k \leq 4) \\
\equiv\ & r > 6n - 6i - k - 1 \wedge i < n \wedge 0 \leq k \leq 5 \quad \equiv: G
\end{aligned}
$$

$$
\begin{aligned}
& \mathsf{WP}[\![\texttt{i == n}]\!](G, N) \\
\equiv\ & \mathsf{WP}[\![\texttt{i == n}]\!](r > 6n - 6i - k - 1 \wedge i < n \wedge 0 \leq k \leq 5, \mathit{true}) \\
\equiv\ & (i \neq n \wedge r > 6n - 6i - k - 1 \wedge i < n \wedge 0 \leq k \leq 5) \vee (i = n) \\
\Longleftarrow\ & i \leq n \wedge r = 6n - 6i - k \wedge 0 \leq k \leq 5 \quad \equiv I
\end{aligned}
$$

$$
\begin{aligned}
& \mathsf{WP}[\![\texttt{r = 6n - 6i - k}]\!](I) \\
\equiv\ & \mathsf{WP}[\![\texttt{r = 6n - 6i - k}]\!](i \leq n \wedge r = 6n - 6i - k \wedge 0 \leq k \leq 5) \\
\equiv\ & i \leq n \wedge 0 \leq k \leq 5 \quad \equiv: F
\end{aligned}
$$

$$
\begin{aligned}
& \mathsf{WP}[\![\texttt{k = 0}]\!](F) \\
\equiv\ & \mathsf{WP}[\![\texttt{k = 0}]\!](i \leq n \wedge 0 \leq k \leq 5) \\
\equiv\ & i \leq n \quad \equiv: E
\end{aligned}
\qquad\qquad
\begin{aligned}
& \mathsf{WP}[\![\texttt{i = 0}]\!](E) \\
\equiv\ & \mathsf{WP}[\![\texttt{i = 0}]\!](i \leq n) \\
\equiv\ & 0 \leq n \quad \equiv: D
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{WP}[\![\texttt{n = -n}]\!](D)\\
\equiv\ &\mathsf{WP}[\![\texttt{n = -n}]\!](0 \le n)\\
\equiv\ &0 \le -n\\
\equiv\ &0 \ge n \quad \equiv: C
\end{aligned}
\qquad
\begin{aligned}
&\mathsf{WP}[\![\texttt{n < 0}]\!](D, C)\\
\equiv\ &\mathsf{WP}[\![\texttt{n < 0}]\!](0 \le n, 0 \ge n)\\
\equiv\ &(n \ge 0 \implies 0 \le n) \wedge (n < 0 \implies 0 \ge n)\\
\equiv\ &\mathit{true} \quad \equiv: B
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{WP}[\![\texttt{n = read()}]\!](B)\\
\equiv\ &\mathsf{WP}[\![\texttt{n = read()}]\!](\mathit{true})\\
\equiv\ &\mathit{true} \quad \equiv: A
\end{aligned}
$$

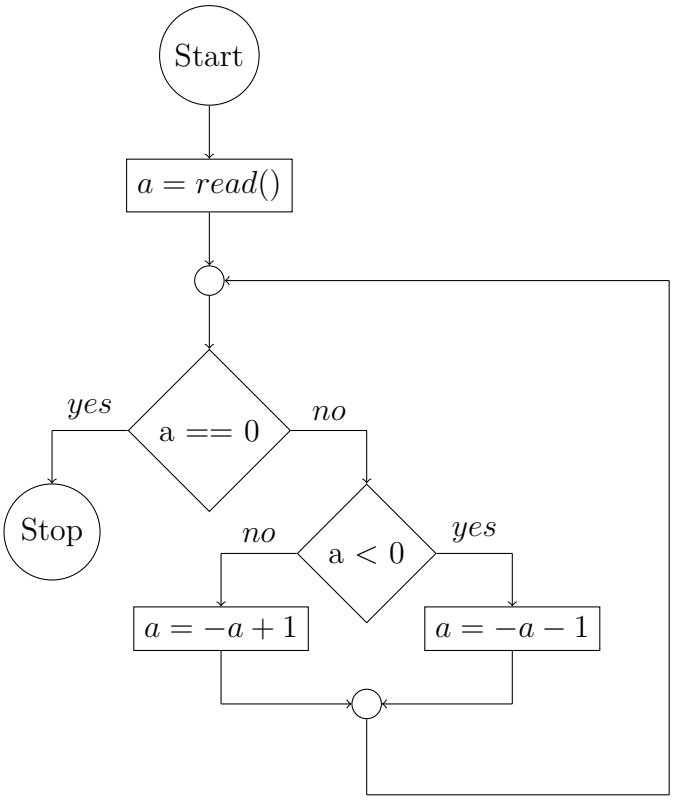
We finally check that for our locally consistent annotations

- $G \equiv r > 6n - 6i - k - 1 \wedge i < n \wedge 0 \le k \le 5 \implies r > 0$ and
- $L \equiv r > 6n - 6i - k \wedge i \le n \wedge 0 \le k \le 5 \implies r > 6n - 6i - k$

does indeed hold. □

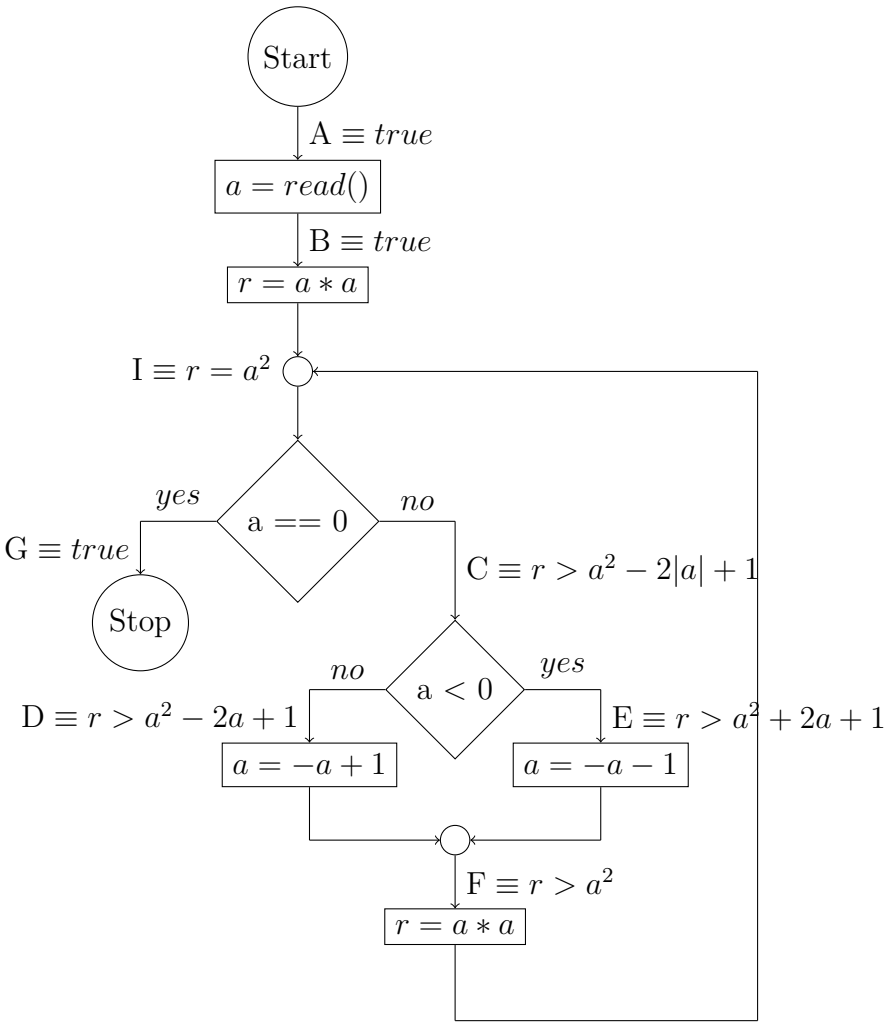
Assignment 4.3 (L) A Wavy Approach

Prove termination of the following program:



Suggested Solution 4.3

We insert the auxiliary variable $r = a * a$ into the control flow graph:



Then, we start with a loop invariant of $r = a^2$ and try to find locally consistent annotations such that

(1) $C \implies r > 0$ and

(2) $F \implies r > a^2$

are satisfied:

$$
\begin{aligned}
&\mathsf{WP}[\![\texttt{r = a * a}]\!](I) \\
\equiv\ &\mathsf{WP}[\![\texttt{r = a * a}]\!](r = a^2) \\
\equiv\ &\mathit{true} \\
\Longleftarrow\ &r > a^2 \quad \equiv: F
\end{aligned}
\qquad
\begin{aligned}
&\mathsf{WP}[\![\texttt{a = -a + 1}]\!](F) \\
\equiv\ &\mathsf{WP}[\![\texttt{a = -a + 1}]\!](r > a^2) \\
\equiv\ &r > a^2 - 2a + 1 \quad \equiv: D
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{WP}[\![\texttt{a = -a - 1}]\!](F) \\
\equiv\ &\mathsf{WP}[\![\texttt{a = -a - 1}]\!](r > a^2) \\
\equiv\ &r > a^2 + 2a + 1 \quad \equiv: E
\end{aligned}
\qquad
\begin{aligned}
&\mathsf{WP}[\![\texttt{a < 0}]\!](D, E) \\
\equiv\ &\mathsf{WP}[\![\texttt{a < 0}]\!](r > a^2 - 2a + 1, r > a^2 + 2a + 1) \\
\equiv\ &(a \geq 0 \wedge r > a^2 - 2a + 1) \vee (a < 0 \wedge r > a^2 + 2a + 1) \\
\equiv\ &r > a^2 - 2|a| + 1 \quad \equiv: C
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{WP}[\![\texttt{a == 0}]\!](C, G) \\
\equiv\ &\mathsf{WP}[\![\texttt{a == 0}]\!](r > a^2 - 2|a| + 1, true) \\
\equiv\ &(a \neq 0 \wedge r > a^2 - 2|a| + 1) \vee a = 0 \\
\Longleftarrow\ &r = a^2 \quad \equiv I
\end{aligned}
$$

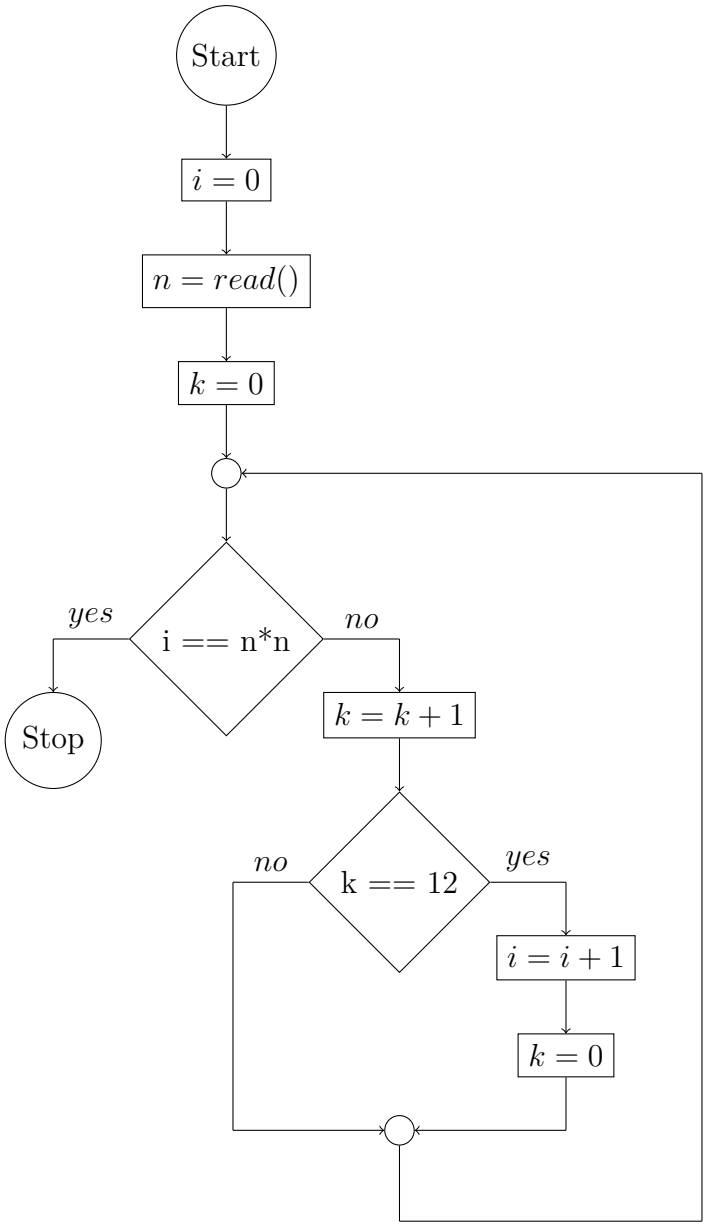
$$
\begin{aligned}
&\mathsf{WP}[\![\texttt{r = a * a}]\!](I) \\
\equiv\ &\mathsf{WP}[\![\texttt{r = a * a}]\!](r = a^2) \\
\equiv\ &true \quad \equiv: B
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{WP}[\![\texttt{a = read()}]\!](B) \\
\equiv\ &\mathsf{WP}[\![\texttt{a = read()}]\!](true) \\
\equiv\ &\texttt{true} \quad \equiv: A
\end{aligned}
$$

□

Assignment 4.4 (H) Counting Twelves [6 Points]

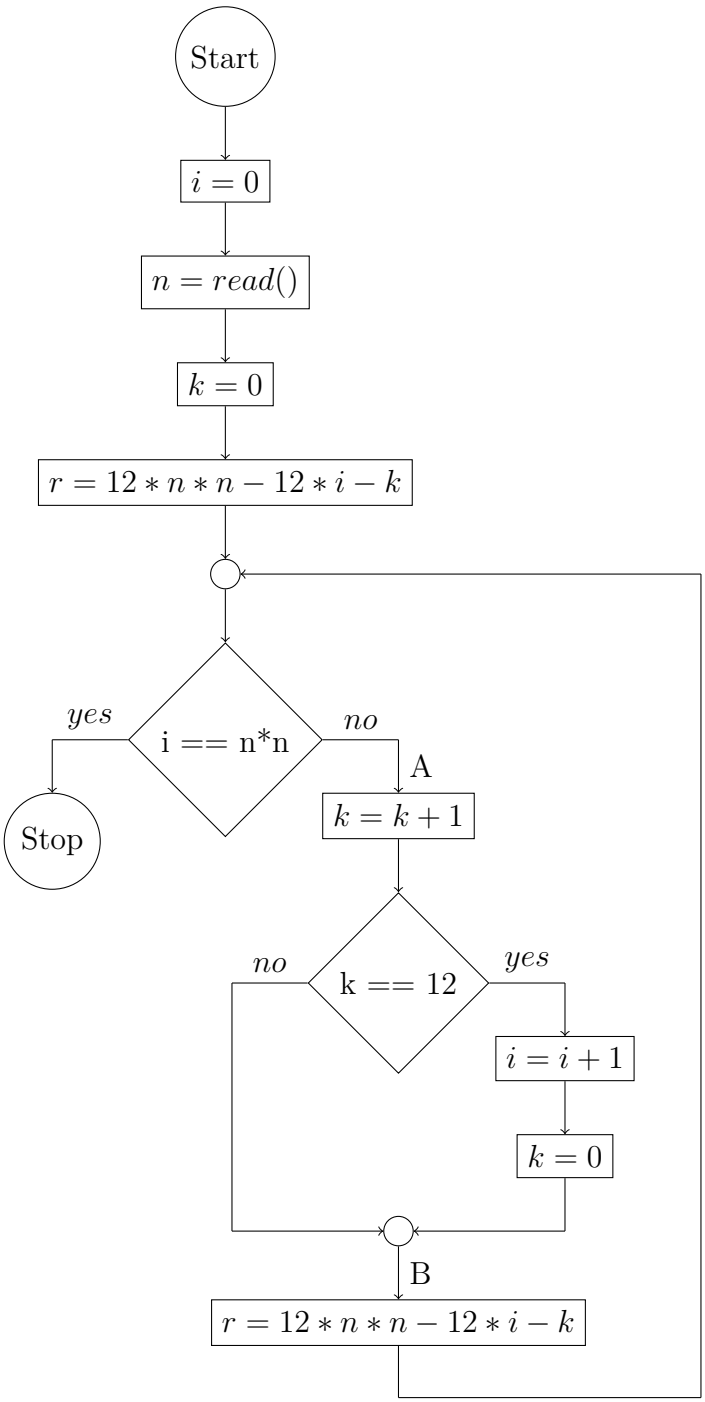
Prove termination of the following program:



Suggested Solution 4.4

We find $r = 12n^2 - 12i - k$ and define our invariant $I :\equiv r = 12n^2 - 12i - k \wedge k < 12 \wedge i \leq n^2$. The goal is to show that

- $A \implies r > 0$ and
- $B \implies r > 12n^2 - 12i - k$


Start
$i = 0$
$n = read()$
$k = 0$
$r = 12 * n * n - 12 * i - k$
yes
i == n*n
no
Stop
A
$k = k + 1$
no
k == 12
yes
$i = i + 1$
$k = 0$
B
$r = 12 * n * n - 12 * i - k$

$$
\begin{aligned}
&\mathsf{WP}[\![\texttt{r = 12 * n * n - 12 * i - k}]\!](I)\\
\equiv\ &\mathsf{WP}[\![\texttt{r = 12 * n * n - 12 * i - k}]\!](r = 12n^2 - 12i - k \land k < 12 \land i \leq n^2)\\
\equiv\ &k < 12 \land i \leq n^2\\
\Longleftarrow\ &r > 12n^2 - 12i - k \land k < 12 \land i \leq n^2 \quad \equiv: B
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{WP}[\![\texttt{k = 0}]\!](B)\\
\equiv\ &\mathsf{WP}[\![\texttt{k = 0}]\!](r > 12n^2 - 12i - k \land k < 12 \land i \leq n^2)\\
\equiv\ &r > 12n^2 - 12i \land i \leq n^2 \quad \equiv: C
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{WP}[\![\texttt{i = i + 1}]\!](C)\\
\equiv\ &\mathsf{WP}[\![\texttt{i = i + 1}]\!](r > 12n^2 - 12i \land i \leq n^2)\\
\equiv\ &r > 12n^2 - 12i - 12 \land i < n^2 \quad \equiv: D
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{WP}[\![\texttt{k == 12}]\!](B, D)\\
\equiv\ &\mathsf{WP}[\![\texttt{k == 12}]\!](r > 12n^2 - 12i - k \land k < 12 \land i \leq n^2, r > 12n^2 - 12i - 12 \land i < n^2)\\
\equiv\ &(k < 12 \land r > 12n^2 - 12i - k \land i \leq n^2) \lor (k = 12 \land r > 12n^2 - 12i - 12 \land i < n^2)\\
\Longleftarrow\ &(k < 12 \land r > 12n^2 - 12i - k \land i < n^2) \lor (k = 12 \land r > 12n^2 - 12i - k \land i < n^2)\\
\equiv\ &r > 12n^2 - 12i - k \land i < n^2 \land k \leq 12 \quad \equiv: E
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{WP}[\![\texttt{k = k + 1}]\!](E)\\
\equiv\ &\mathsf{WP}[\![\texttt{k = k + 1}]\!](r > 12n^2 - 12i - k \land i < n^2 \land k \leq 12)\\
\equiv\ &r > 12n^2 - 12i - k - 1 \land i < n^2 \land k < 12 \quad \equiv: A
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{WP}[\![\texttt{i == n * n}]\!](A, true)\\
\equiv\ &\mathsf{WP}[\![\texttt{i == n * n}]\!](r > 12n^2 - 12i - k - 1 \land i < n^2 \land k < 12)\\
\equiv\ &(i < n^2 \land r > 12n^2 - 12i - k - 1 \land k < 12) \lor i = n^2\\
\Longleftarrow\ &r > 12n^2 - 12i - k - 1 \land k < 12 \land i \leq n^2\\
\Longleftarrow\ &r = 12n^2 - 12i - k \land k < 12 \land i \leq n^2 \quad \equiv I
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{WP}[\![\texttt{r = 12 * n * n - 12 * i - k}]\!](I)\\
\equiv\ &\mathsf{WP}[\![\texttt{r = 12 * n * n - 12 * i - k}]\!](r = 12n^2 - 12i - k\\
&\qquad \land k < 12 \land i \leq n^2)\\
\equiv\ &k < 12 \land i \leq n^2 \quad \equiv: F
\end{aligned}
$$

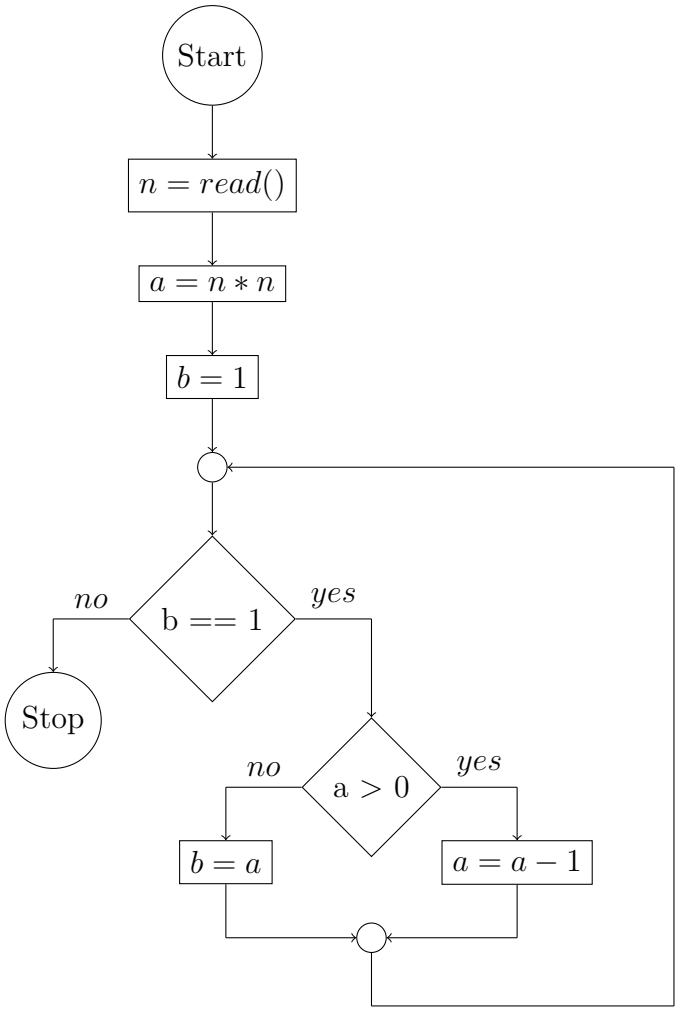
$$
\mathsf{WP}[\![\texttt{k = 0}]\!](F) \ \equiv\ \mathsf{WP}[\![\texttt{k = 0}]\!](k < 12 \land i \leq n^2) \ \equiv\ i \leq n^2 \quad \equiv: G
$$

$$
\mathsf{WP}[\![\texttt{n = read()}]\!](G) \ \equiv\ \mathsf{WP}[\![\texttt{n = read()}]\!](i \leq n^2) \ \equiv\ \forall n.i \leq n^2 \quad \equiv: H
$$

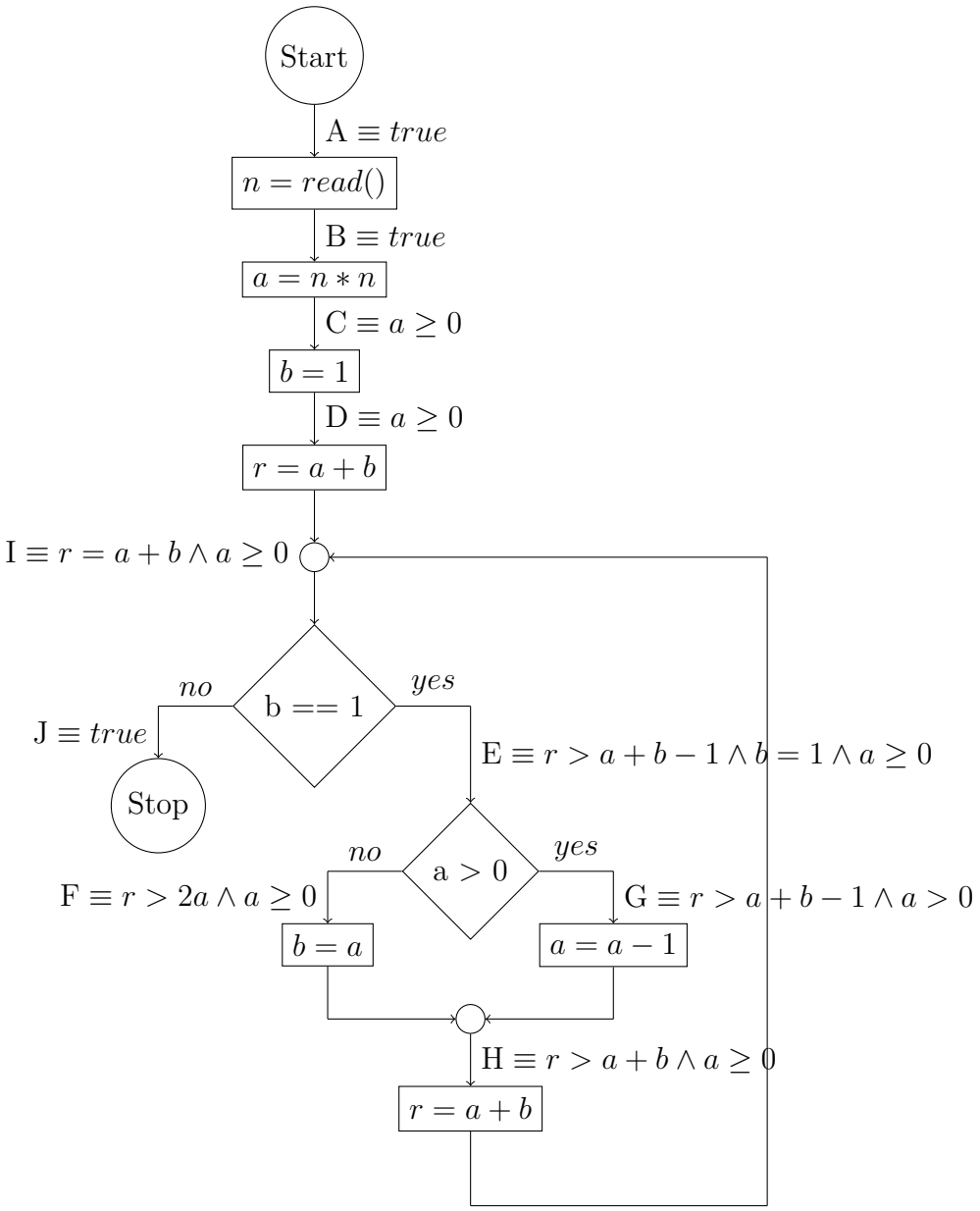
$$
\mathsf{WP}[\![\texttt{i = 0}]\!](H) \ \equiv\ \mathsf{WP}[\![\texttt{i = 0}]\!](\forall n.i \leq n^2) \ \equiv\ \forall n.0 \leq n^2 \ \equiv\ true
$$

Assignment 4.5 (H) aaaaaaaaaab [4 Points]

Prove termination of the following program:



Suggested Solution 4.5



We choose $r = a + b$ and the loop invariant $I :\equiv r = a + b \wedge a \geq 0$ and have to show that $E \implies r > 0$ and $H \implies r > a + b$.

$$
\begin{aligned}
&\mathsf{WP}[\![\texttt{r = a + b}]\!](I)\\
\equiv\ &\mathsf{WP}[\![\texttt{r = a + b}]\!](r = a + b \wedge a \geq 0)\\
\equiv\ &a \geq 0\\
\Longleftarrow\ &r > a + b \wedge a \geq 0 \quad \equiv\!: H
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{WP}[\![\texttt{a = a - 1}]\!](H)\\
\equiv\ &\mathsf{WP}[\![\texttt{a = a - 1}]\!](r > a + b \wedge a \geq 0)\\
\equiv\ &r > a + b - 1 \wedge a > 0 \quad \equiv\!: G
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{WP}[\![\texttt{b = a}]\!](H)\\
\equiv\ &\mathsf{WP}[\![\texttt{b = a}]\!](r > a + b \wedge a \geq 0)\\
\equiv\ &r > 2a \wedge a \geq 0 \quad \equiv\!: F
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{WP}[\![\texttt{a > 0}]\!](F, G)\\
\equiv\ &\mathsf{WP}[\![\texttt{a > 0}]\!](r > 2a \wedge a \geq 0, r > a + b - 1 \wedge a > 0)\\
\equiv\ &(a \leq 0 \wedge r > 2a \wedge a \geq 0) \vee (a > 0 \wedge r > a + b - 1)\\
\Longleftarrow\ &b = 1 \wedge ((a = 0 \wedge r > 2a) \vee (a > 0 \wedge r > a + b - 1))\\
\equiv\ &b = 1 \wedge ((a = 0 \wedge r > a + b - 1) \vee (a > 0 \wedge r > a + b - 1))\\
\equiv\ &r > a + b - 1 \wedge b = 1 \wedge a \geq 0 \quad \equiv\!: E
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{WP}[\![\texttt{b == 1}]\!](J, E)\\
\equiv\ &\mathsf{WP}[\![\texttt{b == 1}]\!](true, r > a + b - 1 \wedge b = 1 \wedge a \geq 0)\\
\equiv\ &b \neq 1 \vee (b = 1 \wedge r > a + b - 1 \wedge a \geq 0)\\
\Longleftarrow\ &r > a + b - 1 \wedge a \geq 0\\
\Longleftarrow\ &r = a + b \wedge a \geq 0 \quad \equiv I
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{WP}[\![\texttt{r = a + b}]\!](I)\\
\equiv\ &\mathsf{WP}[\![\texttt{r = a + b}]\!](r = a + b \wedge a \geq 0)\\
\equiv\ &a \geq 0 \quad \equiv\!: D
\end{aligned}
$$

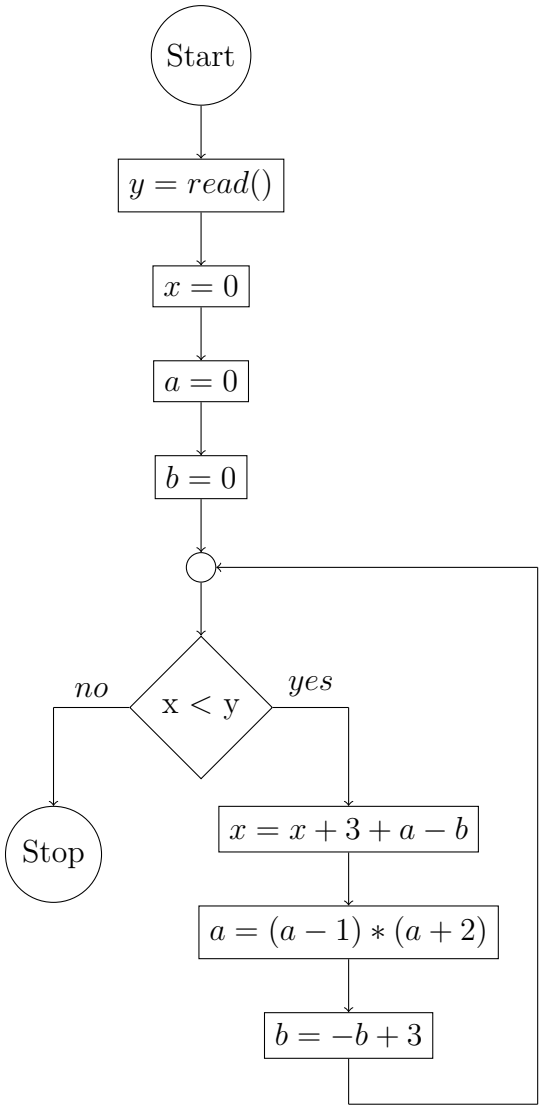
$$
\mathsf{WP}[\![\texttt{b = 1}]\!](D) \ \equiv\ \mathsf{WP}[\![\texttt{b = 1}]\!](a \geq 0) \ \equiv\ a \geq 0 \quad \equiv\!: C
$$

$$
\mathsf{WP}[\![\texttt{a = n * n}]\!](C) \ \equiv\ \mathsf{WP}[\![\texttt{a = n * n}]\!](a \geq 0) \ \equiv\ n * n \geq 0 \ \equiv\ true \quad \equiv\!: B
$$

$$
\mathsf{WP}[\![\texttt{n = read()}]\!](B) \ \equiv\ \mathsf{WP}[\![\texttt{n = read()}]\!](true) \ \equiv\ true \quad \equiv\!: A
$$

Assignment 4.6 (H) Term-ination [10 Points]

Prove that the following program terminates for all inputs:



Hint: Finding a formula for r is non-trivial in this program. Make sure you understand what the program is doing and consider writing down the values of variables for a couple of iterations.
Hint: Keep in mind, that r need not necessarily represent the exact number of remaining loop iterations, but a finite upper limit.

Suggested Solution 4.6

In order to show the individual steps of finding an r, consider this table:

#	a	b	x	$x - b$	$2(x - b)$	$2x + a - b$
0	0	0	0	0	0	0
1	-2	3	3	0	0	1
2	0	0	1	1	2	2
3	-2	3	4	1	2	3
4	0	0	2	2	4	4
5	-2	3	5	2	4	5
6	0	0	3	3	6	6

We try to build up a formula for r. Notice, there are always these $+3$ steps followed by -2 steps in x. If we would subtract 3 from x in every second iteration, the result would be an increasing (although not yet strictly increasing) sequence. Since 3 has to be subtracted from x in exactly these iterations where b has the value 3 and x has to stay unmodified in the iterations where b is 0, we can subtract b, to build the increasing sequence $x - b$. Now, every value is still used twice, so in order to make some room for intermediate steps, we multiply $x - b$ by 2: $2(x - b)$. It remains to add 1 to exactly these iterations where $a = -2$ and $b = 3$. Since $a + b = 1$, we extend our formula to: $2(x - b) + a + b$ (or $2x + a - b$). It remains to subtract this strictly increasing sequence from the total number of iterations, which is $2y$, we define $r = 2y - 2x - a + b$. Notice, that x takes every value greater than 2 twice, and our definition of r represents the number of iterations until it reaches the respective value the second time, although this is not possible in the concrete program. Thus, r is not an exact count for the number of remaining loop iterations, but a safe upper bound and, since we do only need to show that there is a finite number of iterations after which the loop is left, this is a safe over-approximation.

In addition to the value of r, we track information about a and b in the loop variant: Either $a = b = 0$ or $a = -2$ and $b = 3$. The fact that $x < y$ inside the loop is not required in the invariant, since this is implicitly provided by the branch condition:

$$I :\equiv r = 2y - 2x - a + b \land ((a = 0 \land b = 0) \lor (a = -2 \land b = 3))$$

In order to prove termination, we have to show that $A \implies r > 0$ and $B \implies r > 2y - 2x - a + b$.

$$\begin{aligned}
& \mathsf{WP}[\![\texttt{r = 2 * y - 2 * x - a + b}]\!](I) \\
\equiv\ & \mathsf{WP}[\![\texttt{r = 2 * y - 2 * x - a + b}]\!](r = 2y - 2x - a + b \\
& \qquad \land ((a = 0 \land b = 0) \lor (a = -2 \land b = 3))) \\
\equiv\ & (a = 0 \land b = 0) \lor (a = -2 \land b = 3) \\
\impliedby\ & r > 2y - 2x - a + b \land ((a = 0 \land b = 0) \lor (a = -2 \land b = 3)) \quad \equiv: B
\end{aligned}$$

The second termination criteria trivially holds.

$$
\begin{aligned}
&\mathsf{WP}[\![\texttt{b = -b + 3}]\!](B)\\
\equiv\ &\mathsf{WP}[\![\texttt{b = -b + 3}]\!](r > 2y - 2x - a + b\\
&\qquad \wedge ((a = 0 \wedge b = 0) \vee (a = -2 \wedge b = 3)))\\
\equiv\ &r > 2y - 2x - a - b + 3 \wedge ((a = 0 \wedge b = 3) \vee (a = -2 \wedge b = 0)) \quad \equiv: C
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{WP}[\![\texttt{a = (a - 1) * (a + 2)}]\!](C)\\
\equiv\ &\mathsf{WP}[\![\texttt{a = (a - 1) * (a + 2)}]\!](r > 2y - 2x - a - b + 3\\
&\qquad \wedge ((a = 0 \wedge b = 3) \vee (a = -2 \wedge b = 0)))\\
\equiv\ &r > 2y - 2x - ((a-1)(a+2)) - b + 3\\
&\qquad \wedge (((a-1)(a+2) = 0 \wedge b = 3) \vee ((a-1)(a+2) = -2 \wedge b = 0))\\
\equiv\ &r > 2y - 2x - a^2 - a - b + 5\\
&\qquad \wedge ((a^2 + a - 2 = 0 \wedge b = 3) \vee (a^2 + a - 2 = -2 \wedge b = 0))\\
\equiv\ &r > 2y - 2x - a^2 - a - b + 5\\
&\qquad \wedge (((a = 1 \vee a = -2) \wedge b = 3) \vee ((a = -1 \vee a = 0) \wedge b = 0))\\
\Longleftarrow\ &r > 2y - 2x - a^2 - a - b + 5 \wedge ((a = -2 \wedge b = 3) \vee (a = 0 \wedge b = 0)) \quad \equiv: D
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{WP}[\![\texttt{x = x + 3 + a - b}]\!](D)\\
\equiv\ &\mathsf{WP}[\![\texttt{x = x + 3 + a - b}]\!](r > 2y - 2x - a^2 - a - b + 5\\
&\qquad \wedge ((a = -2 \wedge b = 3) \vee (a = 0 \wedge b = 0)))\\
\equiv\ &r > 2y - 2(x + 3 + a - b) - a^2 - a - b + 5 \wedge ((a = -2 \wedge b = 3) \vee (a = 0 \wedge b = 0))\\
\equiv\ &r > 2y - 2x - a^2 - 3a + b - 1 \wedge ((a = -2 \wedge b = 3) \vee (a = 0 \wedge b = 0))\\
\Longleftarrow\ &r > 2y - 2x - a^2 - 3a + b - 1 \wedge x < y \wedge ((a = -2 \wedge b = 3) \vee (a = 0 \wedge b = 0)) \quad \equiv: A
\end{aligned}
$$

We check the first termination criteria by considering the two cases:

- Let $a = 0, b = 0$: $r > 2y - 2x - 1 \wedge x < y \implies r > 0$, since, if $x < y$ then $2y - 2x \geq 2$ and subtracting 1 still guarantees r to be greater than 0.
- Let $a = -2, b = 3$: $r > 2y - 2x + 4 \wedge x < y \implies r > 0$ by the same argument.

$$
\begin{aligned}
&\mathsf{WP}[\![\texttt{x < y}]\!](\mathit{true}, A)\\
\equiv\ &\mathsf{WP}[\![\texttt{x < y}]\!](\mathit{true}, r > 2y - 2x - a^2 - 3a + b - 1 \wedge x < y\\
&\qquad \wedge ((a = -2 \wedge b = 3) \vee (a = 0 \wedge b = 0)))\\
\equiv\ &x \geq y \vee (x < y \wedge r > 2y - 2x - a^2 - 3a + b - 1 \wedge\\
&\qquad ((a = -2 \wedge b = 3) \vee (a = 0 \wedge b = 0))) \quad \equiv: I'
\end{aligned}
$$

Again, we check whether $I \implies I'$:

- Let $a = 0, b = 0$: $r = 2y - 2x \implies r > 2y - 2x - 1$, trivially holds.
- Let $a = -2, b = 3$: $r = 2y - 2x + 5 \implies r > 2y - 2x + 4$, trivially holds.

$$
\begin{aligned}
&\mathsf{WP}[\![\texttt{r = 2 * y - 2 * x - a + b}]\!](I) \\
\equiv\ &\mathsf{WP}[\![\texttt{r = 2 * y - 2 * x - a + b}]\!](r = 2y - 2x - a + b \\
&\qquad \wedge ((a = 0 \wedge b = 0) \vee (a = -2 \wedge b = 3))) \\
\equiv\ &(a = 0 \wedge b = 0) \vee (a = -2 \wedge b = 3) \quad \equiv: E
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{WP}[\![\texttt{b = 0}]\!](E) \\
\equiv\ &\mathsf{WP}[\![\texttt{b = 0}]\!]((a = 0 \wedge b = 0) \vee (a = -2 \wedge b = 3)) \\
\equiv\ &a = 0 \quad \equiv: F
\end{aligned}
$$

$$
\mathsf{WP}[\![\texttt{a = 0}]\!](F) \ \equiv\ \mathsf{WP}[\![\texttt{a = 0}]\!](a = 0) \ \equiv\ \mathit{true} \quad \equiv: G
$$

$$
\mathsf{WP}[\![\texttt{x = 0}]\!](G) \ \equiv\ \mathsf{WP}[\![\texttt{x = 0}]\!](\mathit{true}) \ \equiv\ \mathit{true} \quad \equiv: H
$$

$$
\mathsf{WP}[\![\texttt{y = read()}]\!](H) \ \equiv\ \mathsf{WP}[\![\texttt{y = read()}]\!](\mathit{true}) \ \equiv\ \mathit{true}
$$

All assertions are locally consistent with *true* at the start node and termination requirements are satisfied, thus, we have shown that this program terminates for all inputs. □