IIT CS536: Science of Programming

Homework 6: Loop Bounds, Nondeterminism, Parallelism

Prof. Stefan Muller
TAs: Chaoqi Ma, Zhenghao Zhao

Out: Wednesday, Apr. 6
Due: Monday, Apr. 18, 11:59pm CDT

Updated Apr. 12

This assignment contains 7 written task(s) for a total of 60 points.

Logistics

Submission Instructions

Please read and follow these instructions carefully.

- Submit your homework on Blackboard under the correct assignment by the deadline (or the extended deadline if taking late days).
- You may submit multiple times, but we will only look at your last submission. Make sure your last submission contains all necessary files.
- Email the instructor and TAs ASAP if
  - You submit before the deadline but then decide to take (more) late days.
  - You accidentally resubmit after the deadline, but did not intend to take late days.

  Otherwise, you do not need to let us know if you're using late days; we'll count them based on the date of your last submission.
- Submit your written answers in a single PDF or Word document. Typed answers are preferred (You can use any program as long as you can export a .pdf, .doc or .docx; LaTeX is especially good for typesetting logic and math, and well worth the time to learn it), but *legible* handwritten and scanned answers are acceptable as well.
- Your Blackboard submission should contain only the file with your written answers. Do not compress or put any files in folders.

Collaboration and Academic Honesty

Read the policy on the website and be sure you understand it.

1 Loop Bounds and Proof Outlines

Task 1.1 (Written, 10 points).

Fill the blanks of the following the minimal proof outline, and convert it to a full proof outline. Consider a/b to be truncating integer division: that is, a/b is the greatest integer less than or equal to $\frac{a}{b}$, e.g., $1/2 = 0$. *Updated 4/12*: Remember: you need to consider termination as well as partial correctness; see the while rule for total correctness from class.

$$
\begin{array}{ll}
 & [n > 0] \\
i := n; & \\
\{\textbf{inv}\ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\} & \\
\{\textbf{dec}\ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\} & \\
\mathsf{while}\ (i > 1)\ \{ & \\
\quad i := i\ /\ \overline{2} & \\
\} & [n > 0 \land i = 1]
\end{array}
$$

Task 1.2 (Written, 8 points).

Fill the blanks to complete the following minimal proof outline. You **do not** need to convert it to a full proof outline, just fill in the blanks. Note that one of the blanks is in the program itself!

$$
\begin{array}{ll}
 & [\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \land a[i] \geq 1] \\
\_\_\_\_\_\_\_\_; & \\
k := \overline{0}; & \\
\{\textbf{inv}\ x = 2^k \land \_\_\_\_\_\_\_\_\_\_\_\_\_\_\} & \\
\{\textbf{dec}\ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\} & \\
\mathsf{while}\ (x * 2 \leq a[i])\ \{ & \\
\quad k := k + \overline{1} & \\
\quad x := x * \overline{2} & \\
\} & [x = 2^k \land x \leq a[i] \land a[i] < 2^{k+1}]
\end{array}
$$

2 Nondeterminism

Task 2.1 (Written, 8 points).

Consider two programs, $P_1 \equiv \mathsf{if}\ e_1\ \mathsf{then}\ \{s_1\}\ \mathsf{else}\ \{s_2\}$, and $P_2 \equiv e_1 \to s_1 \,\Box\, e_2 \to s_2$. Fill in the table below, which describes what P_1 and P_2 will do based on various conditions on the state σ.

state σ	P_1	P_2
$\sigma \models e_1 \land e_2$	Executes s_1	
$\sigma \models e_1 \land \neg e_2$		
$\sigma \models \neg e_1 \land e_2$		
$\sigma \models \neg e_1 \land \neg e_2$		

Task 2.2 (Written, 12 points).

a) Calculate $M(\mathsf{havoc}\ i; a[i] = 1, \sigma)$. Show your work.

b) Calculate $M(\mathsf{while}\ \{x > -10 \land x < 10 \to x := x + 1 \,\Box\, x > -10 \land x < 10 \to x := x - 1\}, \{x = 1\})$. You do not need to show work formally, but justify your answer informally in 1-2 sentences.

c) Calculate $M((\mathsf{branch}\ \{x \geq y \to x{:=}x{+}\overline{2} \Box y \geq x \to y{:=}y{+}\overline{1}\}); \mathsf{if}\ (x < y)\ \mathsf{then}\ \{z{:=}y\}\ \mathsf{else}\ \{z{:=}x\}, \{(x = 3, y = 3)\})$. Show your work.

Task 2.3 (Written, 12 points).

Updated 4/12: branch keyword added to a and b to clarify syntax. **Note for this task:** Stefan wrote the sp rule for nondeterministic branches incorrectly in lecture. It should be $sp(p, \textsf{branch}\ \{e_1 \rightarrow s_1\} \square \cdots \square\, e_n \rightarrow s_n) = sp(p \wedge e_1, s_1) \vee \cdots \vee sp(p \wedge e_n, s_n)$. The posted notes are correct.

Calculate the following, and show your work. You **do not** need to fully simplify your answers, but expand all wlp/sp definitions, and perform all substitutions.

a) $wlp(\textsf{branch}\ \{x \geq y \rightarrow max := x \square y \geq x \rightarrow max := y\}, max \geq 0)$

b) $sp(x \geq y, \textsf{branch}\ \{x \geq y \rightarrow y := y + 1 \square y \geq x \rightarrow x := x + 1\})$

c) $wlp(\textsf{havoc}\ x; y := y + 1, y \geq \frac{x}{|x|})$

3 Parallel Programs

Task 3.1 (Written, 10 points).

Consider $s \equiv [\ y := x\ /\ \overline{3} \parallel x := x + \overline{6}; z := x * \overline{2}\ ]$ and state $\sigma = \{x = 18\}$.

a) Draw an evaluation graph for $\langle s, \sigma \rangle$.

b) What is $M(s, \sigma)$? You do not need to show a formal calculation, but explain briefly how you know.

4 One more wrap-up question

Task 4.1 (Written, 0 points).

How long (approximately) did you spend on this homework, in total hours of actual working time? Your honest feedback will help us with future homeworks.