Dr. Matthias Heizmann
Dominik Klumpp

Hand in until 10:00 on June 22, 2020
Discussion: June 22, 2020

Tutorial for Program Verification
Exercise Sheet 10

In this exercise sheet we work on the soundness proof for use the Hoare proof system, and continue with some exercises on the theory of arrays.

Submit your solution by uploading it as PDF in ILIAS.

Exercise 1: Soundness of the Weakening Postcondition Rule 1 Point

Prove that the weakening postcondition rule of the Hoare proof system displayed below is sound.

$$(weakpos)\frac{\{\varphi\}st\{\psi\}}{\{\varphi\}st\{\psi'\}} \text{ if } \psi \models \psi'$$

More precisely, prove the following lemma from the lecture:

> If the Hoare triple $\{\varphi\}st\{\psi\}$ is valid and the side condition $\psi \models \psi'$ is valid, then the Hoare triple $\{\varphi\}st\{\psi'\}$ is valid.

Exercise 2: Soundness of the Composition Rule 2 Bonus Points

Prove that the composition rule of the Hoare proof system displayed below is sound.

$$(compo)\frac{\{\varphi_1\}st_1\{\varphi_2\} \quad \{\varphi_2\}st_2\{\varphi_3\}}{\{\varphi_1\}st_1 st_2\{\varphi_3\}}$$

More precisely, prove the following lemma from the lecture:

> If the Hoare triple $\{\varphi_1\}st_1\{\varphi_2\}$ is valid and the Hoare triple $\{\varphi_2\}st_2\{\varphi_3\}$ is valid, then the Hoare triple $\{\varphi_1\}st_1 st_2\{\varphi_3\}$ is valid.

Exercise 3: Soundness of the Conditional Rule 2 Points

Prove that the conditional rule of the Hoare proof system displayed below is sound.

$$(condi)\frac{\{\varphi \wedge expr\}\ st_1\ \{\psi\} \quad \{\varphi \wedge \neg expr\}\ st_2\ \{\psi\}}{\{\varphi\}\ \texttt{if(}expr\texttt{)\{}st1\texttt{\}else\{}st2\texttt{\}}\ \{\psi\}}$$

More precisely, prove the following lemma from the lecture:

> If the Hoare triple $\{\varphi \wedge expr\}\ st_1\ \{\psi\}$ is valid and the Hoare triple $\{\varphi \wedge \neg expr\}\ st_2\ \{\psi\}$ is valid, then the Hoare triple $\{\varphi\}\ \texttt{if(}expr\texttt{)\{}st1\texttt{\}else\{}st2\texttt{\}}\ \{\psi\}$ is valid.

Exercise 4: Square 2 Points

Find an inductive loop invariant for the while loop of the following program that is strong enough to prove that the program satisfies the given precondition-postcondition pair, i.e., the formulas after `requires` and `ensures`, respectively. Use Ultimate Referee[1] to check your solution.

Alternatively you may also use your own solution to exercise 5 on exercise sheet 8.

```
procedure square(n: int) returns (res: int)
requires n >= 0;
ensures res == n*n;
{
  var i, odd : int;
  i := 0;
  odd := 1;
  res := 0;
  while (i < n) {
    res := res + odd;
    odd := odd + 2;
    i := i + 1;
  }
}
```

Exercise 5: Satisfiability in the Theory of Arrays 2 Points

Determine which of the following FOL formulas is satisfiable in the theory of arrays. If a formula is satisfiable, give a satisfying assignment. You may assume that the arrays have integer indices and values.

(a) $\mathit{select}(a, i) = i \land \mathit{store}(a, i, k) = a \land i \neq k$

(b) $a = \mathit{store}(b, k, v) \land \mathit{select}(a, i) \neq \mathit{select}(b, i) \land \mathit{select}(a, j) \neq \mathit{select}(b, j)$

(c) $b = \mathit{store}(a, k, v) \land \forall i.\ i \neq j \rightarrow \mathit{select}(a, i) = \mathit{select}(b, i)$

You may use an SMT solver to solve this task. To declare an array constant `a`, you can use the SMT-LIB command `(declare-fun a () (Array Int Int))`. The function applications for the *select* function and the *store* function are written as usual, e.g. `(select a i)` and `(store a i v)`.

Exercise 6: Theory of Arrays 2 Points

Formalize the following statements as first order logic formulas.

(a) The array a has the value 0 at every index except at index 5, where the value is 23.

(b) The array a contains no duplicate values between the indices 0 and 10 inclusive.

[1] `https://ultimate.informatik.uni-freiburg.de/?ui=int&tool=referee`