Reasoning About Programs

- Week 5 Assessed PMT -

Induction over Recursively Defined Sets, Relations, and Functions

Answers to be submitted to the SAO by 2pm on Monday 12th Feb

Sophia Drossopoulou and Mark Wheelhouse

Aims To practice induction over inductively defined functions. Also, to practice the discovery of the correct specification of auxiliary functions.

Question

Remember the function $\mathsf{DivMod} : \mathbb{Z} \times \mathbb{Z} \to \mathbb{Z} \times \mathbb{Z}$ defined in terms of the partial function $\mathsf{DM} : \mathbb{Z} \times \mathbb{Z} \times \mathbb{Z} \times \mathbb{Z} \to \mathbb{Z} \times \mathbb{Z}$ from course notes:

```
DivMod :: (Int,Int) -> (Int,Int)
DivMod (m,n) = DM (m,n,0,0)

DM :: (Int,Int,Int,Int) -> (Int,Int)
DM (m,n,cnt,acc)
    | acc+n > m = (cnt,m-acc)
    | otherwise = DM (m,n,cnt+1,acc+n)
```

The function $\mathsf{DivMod}(m, n)$ represents integer division and modulus, i.e.

Assrt_1: $\forall m, n, k1, k2 : \mathbb{N}.\ [\ (k1, k2) = \mathsf{DivMod}(m, n) \to m = k1 * n + k2 \ \land\ k2 < n\ ]$

To prove **Assrt_1** we need to characterize the function $\mathsf{DM}(m, n, cnt, acc)$. We will do this through defining and proving a further assertion, **Assrt_2**, which implies **Assrt_1**.

a) Write out the execution of $\mathsf{DivMod}(7, 3)$. **2 points**

b) Write out assertion **Assrt_2**. **4 points**

Hint The assertion **Assrt_2** must imply that
$\forall m, n, k1, k2 : \mathbb{N}.\ [\ \mathsf{DM}(m, n, 0, 0) = (k1, k2)\ \to\ m = k1 * n + k2\ \land\ k2 < n\ ]$.
But it must be more general than that, ie it must have the form
$\forall m, n, cnt, acc, k1, k2 : \mathbb{N}.\ [\ \mathsf{DM}(m, n, cnt, acc) = (k1, k2)\ \to\ ....\ ]$.

c) Prove that **Assrt_2** $\to$ **Assrt_1**. **4 points**

d) Prove **Assrt_2**. **10 points**

A possible answer:

a: Write out the execution of $\mathsf{DivMod}(7,3)$

$$\begin{array}{lcll} \mathsf{DivMod}(7,3) & = & \mathsf{DM}(7,3,0,0) & \text{definition of } \mathsf{DivMod} \\ & = & \mathsf{DM}(7,3,1,3) & \text{definition of } \mathsf{DM} \text{ – second case} \\ & = & \mathsf{DM}(7,3,2,6) & \text{definition of } \mathsf{DM} \text{ – second case} \\ & = & (2,1) & \text{definition of } \mathsf{DM} \text{ – first case} \end{array}$$

b: Write out **Assrt_2**

The solution we show here was proposed by Benjamin Cunton (2015). It is more succinct than the one Sophia had originally deverloped, but perhaps not that well-suited to imperative programming. We will show the alternative solution below.

Assrt_2: $\forall m, n, acc, cnt, k1, k2 : \mathbb{Z}.$
$$[\ (k1,k2) = \mathsf{DM}(m,n,cnt,acc) \ \to \ m - acc = (k1 - cnt) * n + k2 \ \land \ k2 < n \ ]$$

Informal Explanation: One way of thinking about **Assrt_2** was proposed by Erik Zou (2016): Given input m, n, acc, and cnt, the function will terminate after another $(k1 - cnt)$ calls, the final value for the acc parameter for the last call will be $(k1 - cnt) * n + acc$, and the remainder will be $k2$.

Notice also that even though $MDP(m,n,cnt,acc)$ might not terminate for negative n, this does not affect the validity of **Assrt_2**.

c: Prove that **Assrt_2** $\to$ **Assrt_1**.

Take $m, n, k1, k2 : \mathbb{N}$ arbitrary.
Assume that
(1) $\mathsf{DivMod}(m,n) = (k1,k2)$
We want to show
(α) $m = k1 * n + k2 \ \land \ k2 < n.$

By (1) and definition of DivMod, we obtain
(2) $(k1,k2) = \mathsf{DM}(m,n,0,0)$
Because $m, n, k1, k2 : \mathbb{N}$, we also have that $m, n, k1, k2 : \mathbb{Z}$. Hence **Assrt_2** is applicable, and from (2) we obtain that
(3) $m - 0 = (k1 - 0) * n + k2.$
(4) $k2 < n.$
From (3) and arithmetic we obtain
(5) $m = k1 * n + k2$
From (3) and (5) we obtain α.

d: Prove **Assrt_2**.

We will apply the induction principle described in the notes for the function DM. We will replace predicate $Q(m, n, cnt, acc, k1, k2)$ from the slides by the assertion
$m - acc = (k1 - cnt) * n + k2 \ \wedge \ k2 < n$.

Base Case **4 points**

To Show $\forall m, n, cnt, acc : \mathbb{Z}$.
$[\ acc + n > m \quad \rightarrow \quad m - acc = (cnt - cnt) * n + (m - acc) \ \wedge \ (m - acc) < n \ ]$

Take arbitrary $m, n, cnt, acc : \mathbb{Z}$. Assume that
$(Ass1)$ $acc + n > m$
It remains to show
(α) $m - acc = (cnt - cnt) * n + (m - acc) \ \wedge \ (m - acc) < n$.

The first conjunct of α follows from arithmetic. The second conjunct follows from $(Ass1)$.

Inductive Step **6 points**

Take $n, m, cnt, acc, k1, k2 : \mathbb{Z}$, arbitrary.
Assume that
$(Ass1)$ $acc + n \leq m$
$(Ass2)$ $\mathsf{DM}(m, n, cnt + 1, acc + n) = (k1, k2)$

Inductive Hypothesis $m - (acc + n) = (k1 - (cnt + 1)) * n + k2 \ \wedge \ k2 < n$

To Show $m - acc = (k1 - cnt) * n + k2 \ \wedge \ k2 < n$

We have:

$$
\begin{aligned}
m - acc &= (m - (acc + n)) + n && \text{by arithmetic} \\
&= (k1 - (cnt + 1)) * n + k2 + n && \text{by first conjunct of Inductive Hypothesis} \\
&= (k1 - cnt) * n + k2 && \text{arithmetic}
\end{aligned}
$$

The above, and the second conjunct of Inductive Hypothesis give what was to be shown. This completes the inductive step.

Another possible answer for parts b-d

b: Write out **Assrt_2**

The new version of **Assrt_2** is a bit longer, but better suited to how we would argue if we turned the tail-recursive function DM into a `while`-loop. Note that whenever execution of $\mathsf{DM}(m, n, 0, 0)$ reaches intermediate terms of the form $\mathsf{DM}(m, n, cnt, acc)$, then the accumulator acc holds the value $n * cnt$, i.e. $acc = cnt * n$, and the accumulator never exceeds the value of m, i.e. $acc \leq m$. Therefore, we can define:

Assrt_2': $\forall m, n, acc, k1, k2 : \mathbb{N}$.
$[\, cnt * n = acc \leq m \ \wedge \ (k1, k2) = \mathsf{DM}(m, n, cnt, acc) \quad \rightarrow \quad m = k1 * n + k2 \ \wedge \ k2 < n \ ] \,]$

where we write $cnt * n = acc \leq m$ as a shorthand for $cnt * n = acc \wedge acc \leq m$.

Comparison: **Assrt_2'** is weaker than **Assrt_2**, because it is only concerned with a subset of the possible inputs to $\mathsf{DM}(\_,\_,\_,\_)$. In that sense, the assumption $cnt * n = acc \leq m$ is a precondition to the function. As we will see later, when we turn the function into a loop, the assertion $cnt * n = acc \leq m$ will become the *loop invariant*.

c: Prove that **Assrt_2'** $\rightarrow$ **Assrt_1**.

Take $m, n, k1, k2 : \mathbb{N}$ arbitrary.
Assume that
(1) $\mathsf{DivMod}(m, n) = (k1, k2)$
This, by definition of DivMod implies that
(1) $\mathsf{DM}(m, n, 0, 0) = (k1, k2)$
Because we have that $0 * n = 0 \leq m$, we can apply **Assrt_2'** on (2) and obtain that
(3) $m = k1 * n + k2 \wedge k2 < n$.
This proves (α), and concludes the proof

d: Prove **Assrt_2'**.

For the proof we will apply the induction principle described in the notes for the function DM. We will replace predicate $Q(m, n, cnt, acc, k1, k2)$ from the slides by the assertion
$cnt * n = acc \leq m \rightarrow m = k1 * n + k2 \wedge k2 < n$.

Base Case **4 points**

To Show $\forall m, n, cnt, acc : \mathbb{N}.$
$[\ acc + n > m \rightarrow [\, cnt * n = acc \leq m \rightarrow m = cnt * n + (m - acc) \wedge m - acc < n \,]\ ]$

Take arbitrary $m, n, cnt, acc : \mathbb{N}$.
Assume that
(1) $acc + n > m$
and
(2) $cnt * n = acc$
(3) $acc \leq m$

It remains to show
(α) $m = cnt * n + (m - acc) \wedge m - acc < n$.

From arithmetic we have $m = acc + (m - acc)$, and by applying (2) we get
(4) $m = cnt * n + (m - acc)$.

From (1) and arithmetic we get
(5) $m - acc < n$.

From (4) and (5) we obtain α.

Inductive Step **6 points**

Take $n, m, cnt, acc, k1, k2 : \mathbb{N}$, arbitrary.
Assume that
(1) $acc + n \leq m$
(2) $\mathsf{DM}(m, n, cnt{+}1, acc{+}n) = (k1, k2)$

Inductive Hypothesis $(cnt{+}1) * n = (acc{+}n) \leq\ m\ \rightarrow\ m = k1 * n{+}\ k2\ \wedge\ k2 < n$

To Show $cnt * n = acc \leq m\ \rightarrow\ m = k1 * n{+}k2\ \wedge\ k2 < n$

We assume:
(3) $cnt * n = acc \leq m$
From (3) by arithmetic we obtain
(4) $(cnt + 1) * n = acc + n$
From (4) and (1) we obtain
(5) $(cnt + 1) * n = acc + n \leq m$
By application of the Induction Hypothesis on (5) and (2) we obtain
(6) $m = k1 * n{+}\ k2\ \wedge\ k2 < n$

This completes the inductive step.

Thank you to Benjamin Cunton (2015) for suggesting the first solution, to Krysia Broda, Erik Zou (2016), Jan Matas (2014), for feedback, and to Nicholas Sim (2015) for noticing original discrepancies in the domains of the quantified variables.