Wednesday, 6 April 2022 11.05

C12.1 Verify the claim in Section 12.2 of Lecture 12 that for all $n \geq 0$, $m_0 \in \mathbb{Z}$ the following state-transformation relationship holds:

$$\langle i = n, m = m_0 \rangle$$
$$[\![\textbf{while } i > 0 \textbf{ do } m \leftarrow 2 * m; i \leftarrow i - 1 \textbf{ od}]\!]$$
$$\langle i = 0, m = m_0 * 2^n \rangle$$

(*Hint:* Induction on n.)

C12.2 Verify the following loop invariant claims in Section 12.4 of Lecture 12 hold for any $N \in \mathbb{Z}$:

(i)
$$\{m * 2^i = N\} m \leftarrow 2 * m;\ i \leftarrow i - 1 \{m * 2^i = N\}$$

(ii)
$$\{m * 2^i = N\}$$
$$\textbf{while } i > 0 \textbf{ do } m \leftarrow 2 * m;\ i \leftarrow i - 1 \textbf{ od}$$
$$\{m * 2^i = N\}$$

You don't need to go back to the detailed semantics of the programming language here; commonsense mathematical reasoning suffices.

- A program S then induces a semi-computable relation[1] $[\![S]\!]$ between states:
  $\omega[\![S]\!]\omega' \iff$ executing S in initial state ω halts and results in final state ω'.
- The state-transformation relation $[\![S]\!]$ can be taken as the semantical "meaning" of program S.

- The semantics of the composite statement is obtained by simple relation composition:
  $\omega[\![S_1; S_2; \ldots; S_k]\!]\omega' \iff \omega([\![S_1]\!] \circ [\![S_2]\!] \circ \cdots \circ [\![S_k]\!])\omega'$

For a given initial state $\omega = \langle i = n, m = m_0 \rangle$ the statement $[\![S]\!]$ transforms it to state $\omega' = \langle i = 0, m = m_0 * 2^n \rangle$

By induction on n:

Base case, $n = 0$:

Then for $\omega = \langle i = n, m = m_0 \rangle$ it holds that $m = m_0 * 2^0$ and $i = n = 0$. So the claim holds for $i = 0$.

Induction assumption: Assume that it holds for some $k \in \mathbb{N}$.

Induction step: Let $\omega = \langle i = k+1, m = m_0 \rangle$. Rewrite the while loop S as $S_1; S_2$; where S_1 is a while loop for $i = k+1$ to $i = 1$ and S_2 is an assignment as follows

$\omega = \langle i = k+1, m = m_0 \rangle$

$S = [\![\{ \text{while } i > 1 \text{ do } m \leftarrow 2 * m, i \leftarrow i - 1 \text{ od}; \} S_1$
$\quad m \leftarrow 2 * m, i \leftarrow i - 1]\!] \} S_2$

By induction assumption: after statement S_1 $m = m_0 * 2^k$ and $i = 1$ and so after S_2 we have $m = m_0 * 2^k \cdot 2 = m_0 * 2^{k+1}$ and $i = 0$.

And therefore $\omega' = \langle i = 0, m = m_0 * 2^{k+1} \rangle$.