Concurrency theory
Exercise sheet 11

TU Braunschweig
Sebastian Muskalla, Prakash Saivasan
Winter term 2017/18

Out: January 17 Due: January 23

Submit your solutions until Tuesday, January 23, during the lecture.

Exercise 1

Consider two traces $\tau = \alpha.a.b.\gamma$ and $\tau' = \alpha'.a.\beta.b.\gamma'$ where thread(c) $\neq$ thread(a) and thread(c) $\neq$ thread(b) for all c in β. Prove the following:

$$\text{If } a \rightarrow_{\mathsf{hb}} b \text{ in } \mathsf{Tr}_{\mathsf{TSO}}(\tau) \text{ then } a \rightarrow_{\mathsf{hb}}^{+} b \text{ in } \mathsf{Tr}_{\mathsf{TSO}}(\tau')$$

Exercise 2

Consider the following program implementing an instance of the **non-blocking write** protocol by H. Kopetz and J. Reisinger:

ℓ_1 : $h \leftarrow mem[g]$; goto ℓ_2	ℓ_9 : $h \leftarrow mem[g]$; goto ℓ_{10}
ℓ_2 : $mem[g] \leftarrow h + 1$; goto ℓ_3	ℓ_{10} : $mem[g] \leftarrow h + 1$; goto ℓ_{11}
ℓ_3 : $mem[x] \leftarrow 42$; goto ℓ_4	ℓ_{11} : $mem[x] \leftarrow 43$; goto ℓ_{12}
ℓ_4 : $mem[g] \leftarrow h + 2$; goto ℓ_5	ℓ_{12} : $mem[g] \leftarrow h + 2$;
ℓ_5 : $r \leftarrow mem[g]$; goto ℓ_6	
ℓ_6 : $v \leftarrow mem[x]$; goto ℓ_7	
ℓ_7 : $s \leftarrow mem[g]$; goto ℓ_8	
ℓ_8 : assert $r \neq s \vee r$ is odd; goto ℓ_5	
ℓ_8 : assert $r = s \wedge r$ is even;	

Note that there are two instructions labeled by ℓ_8. Assume that when executing goto ℓ_8, the execution non-deterministically jumps to any of them.

Prove that the program is not robust under TSO. Initially assume $mem[g] = 0$ and $g \neq x$.

Exercise 3

Consider a computation $\tau = \tau_1.act_1.\tau_2 \in \mathsf{C}_{\mathsf{SC}}(P)$ where for all act_2 in τ_2 we have $act_1 \rightarrow_{hb}^{*} act_2$. Show that the computation $\tau.act$ satisfies $act_1 \rightarrow_{hb}^{*} act$ if and only if

1. there is an action act_2 in $act_1.\tau_2$ with $thread(act_2) = thread(act)$, or
2. act is a load whose address is stored in $act_1.\tau_2$, or
3. act is a store (with issue) whose address is loaded or stored in $act_1.\tau_2$.