SE 4367 Homework 7

Rachel LaViola
2/21/2019

Break into mutually singular components:

Pr: $\underbrace{(a+b)}_{\text{singular}}\underbrace{(c+!cd)}_{\text{non-singular}}$ ✓ already mutually singular

singular ↑ use BOR-CSET

non-singular ↑ use MI-CSET

BOR-CSET

① Generate AST Tree

$S_{N_1}^t = \{(t)\}$ $S_{N_2}^t = \{(t)\}$

$S_{N_1}^f = \{(f)\}$ $S_{N_2}^f = \{(f)\}$

N3 is an OR node:

$S_{N_3}^t = (\{(t)\} \times \{(f)\}) \cup (\{(f)\} \times \{(t)\})$

$= \{(t,f)\} \cup \{(f,t)\}$

$= \{(t,f),(f,t)\}$

$S_{N_3}^f = \{(f)\} \otimes \{(f)\}$

$= \{(f,f)\}$

Constraint set for BOR Portion: $\{(t,f),(f,t),(f,f)\}$

MI-CSET

① Derive constraints to make each e in $E = \overset{e_1}{c} + \overset{e_2}{!cd}$ true:

$T_{e_1} = \{(t,t),(t,f)\}$

$T_{e_2} = \{(F,t)\}$

There are no duplicates, so skip step 2.

③ Select an element from each set:

$S_E^t = \{(t,t),(f,t)\}$

④ For each term in E, complement each literal one at a time:

$e_1 = !c$

$e_2^1 = cd$ $e_2^2 = !c!d$

⟶

(4.2) Derive constraints to make each $e_{complement}$ true:

$Fe_1 = \{(f,t), (f,f)\}$!c

$Fe_2^1 = \{(t,t)\}$ cd

$Fe_2^2 = \{(f,f)\}$!c!d

(5) Remove terms from F_e that appear in any set of T:

$FSe_2^1 = \{(f,f)\}$

$FSe_2^1 = \{\}$

$FSe_2^2 = \{(f,f)\}$

(6) Create a minimal set that has elements that come from each FS.

$S_E^f = \{(f,f)\}$

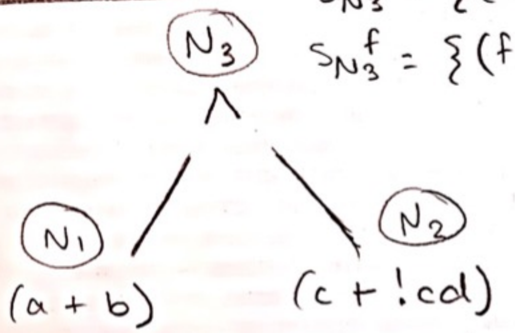
(7) $S_E = S_E^t \cup S_E^f$

$S_E = \{(t,t), (f,t), (f,f)\}$

BOR MI-CSET

$S_{N3}^t = \{(t,t,t,t), (f,t,f,t)\}$

$S_{N3}^f = \{(f,f,t,t), (t,f,f,f)\}$



$S_{N1}^t = \{(t,f), (f,t)\}$

$S_{N1}^f = \{(f,f)\}$

$S_{N2}^t = \{(t,t), (f,t)\}$

$S_{N2}^f = \{(f,f)\}$

N_3 is an AND node:

$$S_{N3}^{t} = \{(t,t),(f,t)\} \otimes \{(t,t),(f,t)\}$$

$$= \{(t,t,t,t),(f,t,f,t)\}$$

$$S_{N3}^{f} = (\{(f,f)\} \times \{(t,t)\}) \cup (\{(t,f)\} \times \{(f,f)\})$$

$$= \{(f,f,t,t)\} \cup \{(t,f,f,f)\}$$

$$= \{(f,f,t,t),(t,f,f,f)\}$$

Constraint set for BOR MI-CSET:

$$S_{N4} = \{(t,t,t,t),(f,t,f,t),(f,f,t,t),(t,f,f,f)\}$$

Generate a test set for pr:

	a	b	c	d	Testcase <a, b, c, d>
t_1	t	t	t	t	<true, true, true, true>
t_2	f	t	f	t	<false, true, false, true>
t_3	f	f	t	t	<false, false, true, true>
t_4	t	f	f	f	<true, false, false, false>