` *FILE: bools.x*
` *PURPOSE: test all logical operators*
` *METHOD: The X assert if boolexp ? fi is used to check answers.*
pass ← 0; ` *none yet*
` *test literal constant, store and fetch*
b2, b1 ← false, true;
if b1 ∧ ¬b2 ⇒ **fi**;
pass ← pass+2;
` *test multiple assign, —*
b6, b5, b4, b3 ← b1 ∨ b1, true ∨ b1, b1 ∨ false, true ∨ false;
if b3 ∧ b4 ∧ b5 ∧ b6 ⇒ **fi**;
pass ← pass+4;
` *test multiple assign,*
b10, b9, b8, b7 ← b1 ∧ b1, true ∧ b1, b1 ∧ false, true ∧ false;
if ¬b7 ∧ ¬b8 ∧ b9 ∧ b10 ⇒ **fi**;
pass ← pass+4;
` *test bool*
b13, b12, b11 ← ¬(¬(¬b1)), ¬(¬b1), ¬b1;
`*xx1:=b11;*
`*xx2:=b12;*
`*xx3:=b13;*
`*xx4:= b11;*
`*xx5:= b13;*
`*xx6:= b11 b12 b13;*
if ¬b11 ∧ b12 ∧ ¬b13 ⇒ **fi**;
pass ← pass+3;
` *test int relations*
b16, b15, b14 ← 1 = 2, 1 ≤ 2, 1 < 2;
b19, b18, b17 ← 1 < 2, 1 ≥ 2, 1 ≠ 2;
if b14 ∧ b15 ∧ ¬b16 ∧ b17 ∧ ¬b18 ∧ ¬b19 ⇒ **fi**;
pass ← pass+6;
b16, b15, b14 ← 1 = 1, 1 ≤ 1, 1 < 1;
b19, b18, b17 ← 1 < 1, 1 ≥ 1, 1 ≠ 1;
if ¬b14 ∧ b15 ∧ b16 ∧ ¬b17 ∧ b18 ∧ ¬b19 ⇒ **fi**;
pass ← pass+6;
` *test real relations*
b15, b14 ← 1.1 ≤ 2.2, 1.1 < 2.2;
b19, b18 ← 1.1 < 2.2, 1.1 ≥ 2.2;
if b14 ∧ b15 ∧ ¬b18 ∧ ¬b19 ⇒ **fi**;
pass ← pass+4;
b15, b14 ← 1.1 ≤ 1.1, 1.1 < 1.1;
b19, b18 ← 1.1 < 1.1, 1.1 ≥ 1.1;
if ¬b14 ∧ b15 ∧ b18 ∧ ¬b19 ⇒ **fi**;
pass ← pass+4;
b21, b20 ← 2.1 ≤ 1.1, 2.1 < 1.1;
b23, b22 ← 2.1 < 1.1, 2.1 ≥ 1.1;
if ¬b20 ∧ ¬b21 ∧ b22 ∧ b23 ⇒ **fi**;
pass ← pass+4;
` *parens*
b27, b26 ← (¬(¬(¬false))), (((((((true)))))));
if b26 ∧ ¬b27 ⇒ **fi**;
pass ← pass+2;
` *precedence*
b31, b30, b29, b28 ← ¬b1 ∨ b1 ∧ ¬b1, ¬b1 ∨ ¬b1 ∧ b1, ¬b1 ∨ b1 ∧ b1, b1 ∨ b1 ∧ b1;
if b28 ∧ b29 ∧ ¬b30 ∧ ¬b31 ⇒ **fi**;
pass ← pass+4;