

\$SPAD/src/input westerboolean.input

Michael Wester

December 16, 2014

**Abstract**

These problems come from the web page  
[http://math.unm.edu/~wester/cas\\_review.html](http://math.unm.edu/~wester/cas_review.html)

# Contents

1 Boolean

3

— \* —

```
)set break resume
)set messages autoload off
)set streams calculate 7
)sys rm -f westerboolean.output
)spool westerboolean.output
)clear all
```

—————

## 1 Boolean

Simplify logical expressions =, false

— \* —

```
--S 1 of 11
true and false
--R
--R
--R (1) false
--R
--E 1
```

Type: Boolean

—————

Argument number 1 to “or” must be a Boolean.

— \* —

```
--S 2 of 11
x : Boolean
--R
--R
--E 2
```

Type: Void

```
--S 3 of 11
x or (not x)
--R
--R
--R x is declared as being in Boolean but has not been given a value.
--E 3
```

—————

=> *xory*

— \* —

```

--S 4 of 11
y : Boolean
--R
--R
--R                                         Type: Void
--E 4

```

```

--S 5 of 11
x or y or (x and y)
--R
--R
--R x is declared as being in Boolean but has not been given a value.
--E 5

```

$$x \quad \frac{\quad}{\quad * \quad}$$

```

--S 6 of 11
xor(xor(x, y), y)
--R
--R
--R x is declared as being in Boolean but has not been given a value.
--E 6

```

$$=, \frac{\quad}{\quad * \quad} [\text{not } (w \text{ and } x)] \text{ or } (y \text{ and } z)$$

```

--S 7 of 11
w : Boolean
--R
--R
--R                                         Type: Void
--E 7

```

```

--S 8 of 11
z : Boolean
--R
--R
--R                                         Type: Void
--E 8

```

```

--S 9 of 11
implies(w and x, y and z)
--R
--R
--R w is declared as being in Boolean but has not been given a value.
--E 9

```

$$\frac{\quad}{\quad}$$

```

=> (x and y) or [not (x or y)]
x iff y
=> false

```

— \* —

```

--S 10 of 11
x and 1 > 2
--R
--R
--R x is declared as being in Boolean but has not been given a value.
--E 10

)clear properties w x y z

```

Quantifier elimination: See Richard Liska and Stanly Steinberg, “Using Computer Algebra to Test Stability”, draft of September 25, 1995, and Hoon Hong, Richard Liska and Stanly Steinberg, “Testing Stability by Quantifier Elimination”, “Journal of Symbolic Computation”, Volume 24, 1997, 161–187.

```

=> (a > 0 and b > 0 and c > 0) or (a < 0 and b < 0 and c < 0)
[Hong, Liska and Steinberg, p. 169]
forAll y in C {implies(a*y**2 + b*y + c = 0, real(y) < 0)}
=> v > 1 [Liska and Steinberg, p. 24]
thereExists w in R suchThat _
{v > 0 and w > 0 and -5*v**2 - 13*v + v*w - w > 0}
=> a^2 <= 1/2 [Hoon, Liska and Steinberg, p. 174]
forAll c in R _
{implies(-1 <= c <= 1, a**2*(-c**4 - 2*c**3 + 2*c + 1) + c**2 + 2*c + 1 <= 4)}
=> v > 0 and w > |W| [Liska and Steinberg, p. 22]
forAll y in C _
{implies(v > 0 and y**4 + 4*v*w*y**3 + 2*(2*v**2*w**2 + w**2 + W**2)*y**2 _
+ 4*v*w*(w**2 - W**2) _
+ (w**2 - W**2)**2 = 0, real(y) < 0)}
This quantifier free problem was derived from the above example by QEPCAD
=> v > 0 and w > |W| [Liska and Steinberg, p. 22]

```

— \* —

```

--S 11 of 11
v > 0 and 4*w*v > 0 and 4*w*(4*w**2*v**2 + 3*W**2 + w**2) > 0 _
and 64*w**2*v**2*(w**2 - W**2)*(w**2*v**2 + W**2) > 0 _
and 64*w**2*v**2*(w**2 - W**2)**3*(w**2*v**2 + W**2) > 0
--R
--R
--R (6) true
--R
Type: Boolean

```

--E 11

---

=> B < 0 and a b > 0 [Liska and Steinberg, p. 49 (equation 86)]  
hereExists y in C, thereExists n in C, thereExists e in R suchThat \_  
{real(y) > 0 and real(n) < 0 and y + A\*%i\*e - B\*n = 0 and a\*n + b = 0}

\_\_ \* \_\_

)spool  
)lisp (bye)

---

## References

- [1] nothing