

# 99 questions/Solutions/48

## From HaskellWiki

< 99 questions | Solutions

(\*\*) Truth tables for logical expressions (3).

Generalize problem P47 in such a way that the logical expression may contain any number of logical variables. Define `table/2` in a way that `table(List,Expr)` prints the truth table for the expression `Expr`, which contains the logical variables enumerated in `List`.

```
import Control.Monad (replicateM)

-- functions as in solution 46
infixl 4 `or`
infixl 4 `nor`
infixl 5 `xor`
infixl 6 `and`
infixl 6 `nand`
infixl 3 `equ` -- was 7, changing it to 3 got me the same results as in the original questio.

tablen :: Int -> ([Bool] -> Bool) -> IO ()
tablen n f = mapM_ putStrLn [toStr a ++ " => " ++ show (f a) | a <- args n]
  where args n = replicateM n [True, False]
        toStr = unwords . map (\x -> show x ++ space x)
        space True = " "
        space False = ""
```

Retrieved from "[https://wiki.haskell.org/index.php?title=99\\_questions/Solutions/48&oldid=57452](https://wiki.haskell.org/index.php?title=99_questions/Solutions/48&oldid=57452)"

Category:

- Programming exercise spoilers

- 
- This page was last modified on 18 January 2014, at 19:51.
  - Recent content is available under a simple permissive license.