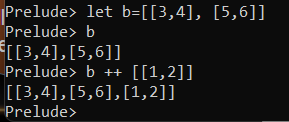
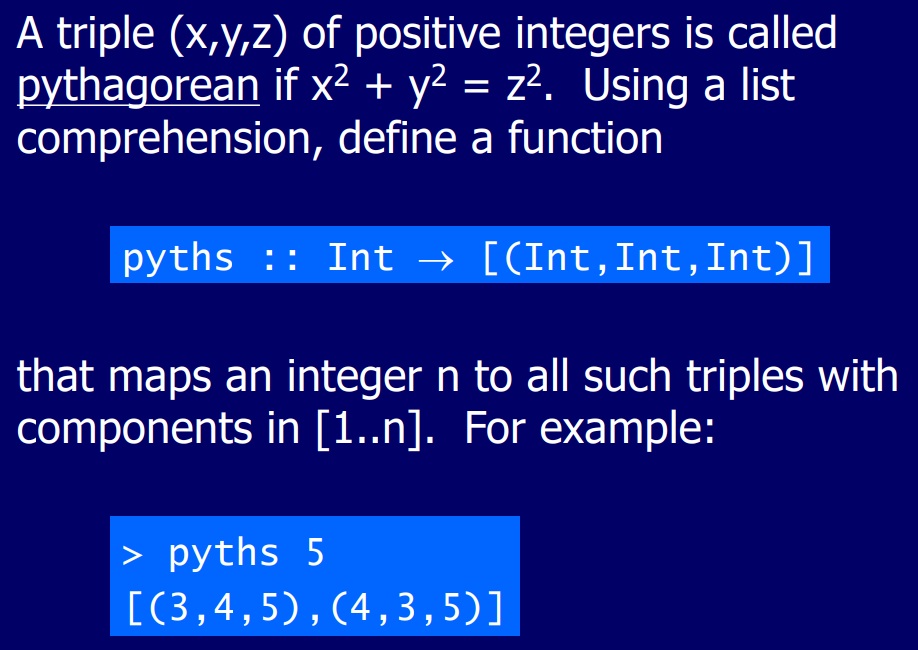
1. <http://zvon.org/other/haskell/Outputprelude/FF_o.html>

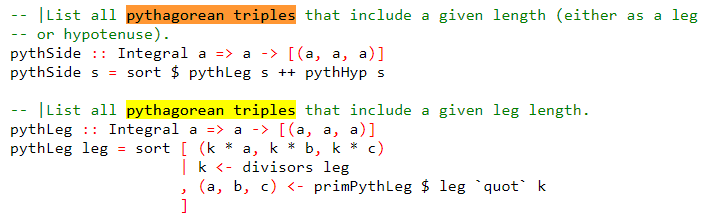
(&&) performs the *and* operation. Given two boolean values, it evaluates to True if both the first and the second are True, and to False otherwise.

1. .



1. .

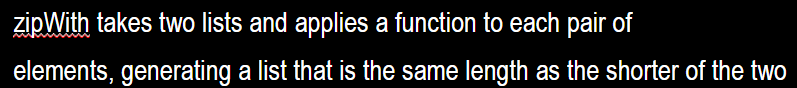




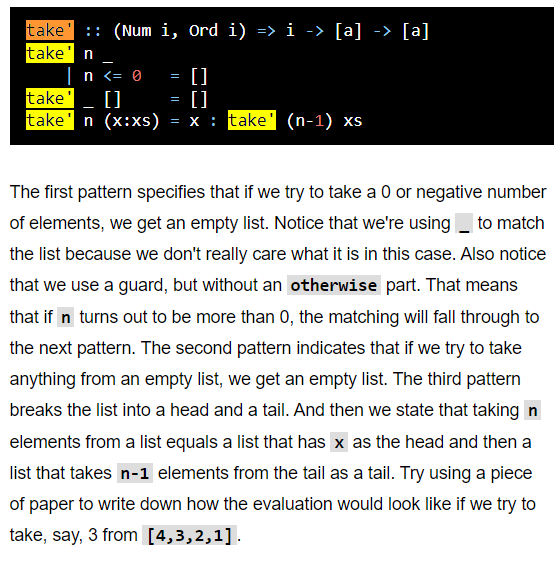
1. zipWith :: (a → b → c) → [a] → [b] → [c ]

-- tail xs removes the first element from the list xs and returns the remainder.

-- Thus, tail [1, 2, 3] equals [2, 3]. tail errors if the list is empty.

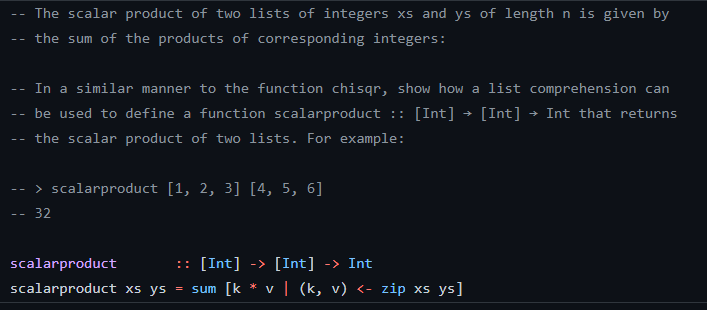


1. no

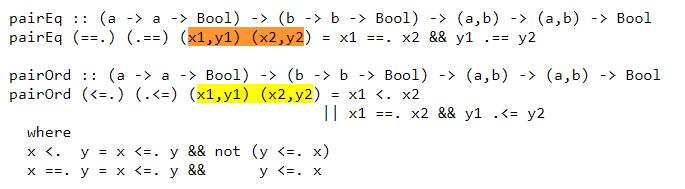


* The underscore symbol \_ is a wildcard pattern that matches any argument value.

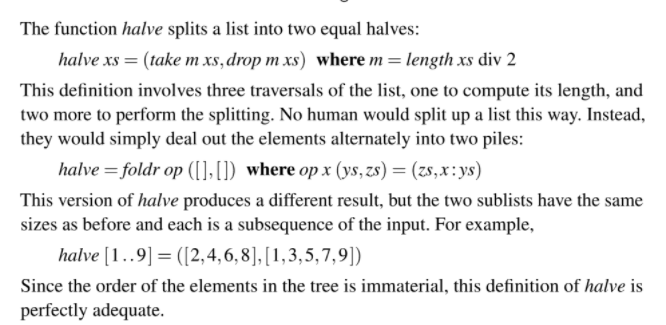
1. .



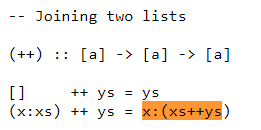
1. .



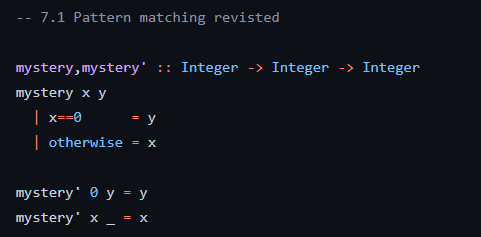
1. .



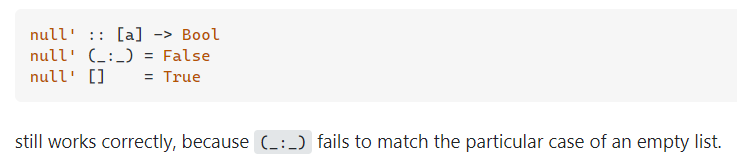
1. .



1. .



1. .



* <https://stackoverflow.com/questions/27941690/implementation-of-null-function>
* And (\_:\_) specifies that the list must have at least one element or the match will fail