Homework 1

BJ Bloemker (A20372354)

1. f x = head(reverse x)
2. head\_repeats n x = take n x == take n (drop n x)
3. The function won't compile if it's definitions don't have the same number of arguments. Based on the first definition, the implication is if both input arguments are 0 then the function should return 1. The following implements such change:  
     
   :{  
   f 0 0 = 1  
   f x n = x \* x \* n  
   :}  
     
   The resulting function is a curried function.
4. swap\_ends x = drop ((length x)-1)x ++ take ((length x)-2)(drop 1 x) ++ take 1 x
5. ++ only takes a single parameter. The definition implies that it takes two parameters which is incorrect. Not changing the right hand side would be invalid because there are 3 parameters with only two unary operators.
6. The num class does not contain the divisor sign, /. This can be fixed by changing Num to Fractional:  
     
   :{  
   f :: (Fractional a) => a -> [a]  
   f x = [x / 2]  
   :}