1. This week's Problem of the Week in Math is described as follows:

> *There are thirty positive integers less than 100 that share a certain property. Your friend, Blake, wrote them down in the table to the left. But Blake made a mistake! One of the numbers listed is wrong and should be replaced with another. Which number is incorrect, what should it be replaced with, and why?*

The numbers are listed below.

6	10	14	15	21
22	26	33	34	35
38	39	46	51	55
57	58	62	65	69
75	77	82	85	86
87	91	93	94	95

Use the fact that the "certain" property is that these numbers are all supposed to be the product of *unique* prime numbers to find and fix the mistake that Blake made.
Reminder: Code your solution in an `R` script and copy it over to this `.Rnw` file.
Hint: You may find the `%in%` operator and the `setdiff()` function to be helpful.

Solution:

```
#needed package:
#install.packages("numbers")
library("numbers")

#list of numbers to check
provided.nums <- c(6, 10, 14, 15, 21,
                   22, 26, 33, 34, 35,
                   38, 39, 46, 51, 55,
                   57, 58, 62, 65, 69,
                   75, 77, 82, 85, 86,
                   87, 91, 93, 94, 95)

#Function to determine which numbers are products of unique primes
#Function takes in a number and returns true if num is a product of unique
#primes and false otherwise
get_valid_numbers <- function(num){
  factors <- primeFactors(num) #get prmime factors of the num
  unique.factors <- unique(factors) #get unique factors
  if (length(unique.factors) ==1){ #if the number contains only 1 prime
    return(FALSE)
  }
  return(length(factors) == length(unique.factors))
}

num.to.check <- c(1:99)
valid.nums <- c()
#check every number if it has unique factors
#if true, add to the list of valid numbers
for (i in 1:length(num.to.check)){
  if (get_valid_numbers(num.to.check[i])){
    valid.nums <- c(valid.nums, num.to.check[i])
  }
}
#numbers that are valid but not in provided list
valid.not.included.nums <- setdiff(valid.nums, provided.nums)
#numbers that are in provided list but not valid
wrong.num <- setdiff(provided.nums, valid.nums)

#get index of the wrong number
wrong.index <- which(provided.nums==wrong.num)
#get the number after wrong
next.index <- wrong.index+1
next.after.wrong = 99
if (next.index <= length(provided.nums)){ #check bounds
  next.after.wrong = provided.nums[next.index]
}
#get the number before wrong
previous.index <- wrong.index-1
```

```
prev.before.wrong = 1
if (previous.index > 0){
  prev.before.wrong = provided.nums[previous.index]
}
current.to.replace = 0 #placeholder for the number that will replace the wrong one
#replace the wrong number with a valid product of unique prime numbes
for (i in 1:length(valid.not.included.nums)){
  #iterate the valid numbers until we find the one in bounds of previous and next
  if (valid.not.included.nums[i] > prev.before.wrong && valid.not.included.nums[i] < next.after.wrong){
    current.to.replace = valid.not.included.nums[i]
  }
}
#replace the wrong number with the correct one
provided.nums[wrong.index] = current.to.replace
```

The wrong number was 75 and it was replaced with 74. 74 in larger than the number that came before 75 (which is 69) and it is smaller than the number that came after 75 (which is 77). Moreover, 74 is a product of unique prime numbers.

Correct list of numbers: 6, 10, 14, 15, 21, 22, 26, 33, 34, 35, 38, 39, 46, 51, 55, 57, 58, 62, 65, 69, 74, 77, 82, 85, 86, 87, 91, 93, 94, 95.