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:

```
prime_factors = function(n){ #returns a vector of prime factors of n
  primefactors = c() #stores prime factors of n
  i = 2 #initialize i to 2 because everything is divisible by 1, and we don't care about that
  while(n>1){
    while(n %% i == 0){ #if i is a factor
      n = n/i #divides by factor to potentially get other factors of n
      primefactors = append(primefactors, i) #add i to vector of prime factors
    }
    i=i+1 #find next factor
    if(i*i>n){ #checks that n has no more prime divisors
      if(n>1){ #if n=1, we can break
        primefactors = append(primefactors, n) #append the prime n to vector of prime factors
      }
      break #if n=1, we are done
    }
  }
  return(primefactors) #return vector of all prime factors (includes duplicates)
}


is_prime = function(n){ #checks if n is prime
  for (i in 3:n-1){
    if(n%%i == 0){ #checks if divisible by i
      return(FALSE)
    }
  }
  return (TRUE) #not divisible by anything other than itself and 1
}

correct = c() #empty vector that will be a vector of products of unique prime numbers
for(i in 3:99){
  primefactors = prime_factors(i) #gets prime factors of i
  if(length(primefactors)==length(unique(primefactors))  && length(primefactors) == 2){ #checks if they are all unique
    if(!is_prime(i)){ #can't be prime because it must be a product of unique prime numbers
      correct = append(correct, i) #add i if so
    }
  }
}

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) #set of numbers Blake wrote down

incorrect = setdiff(nums, correct) #number that is in Blake's set but not my correct numbers vector
replacement = setdiff(correct, nums) #number that is in my correct vector but not Blake's set of numbers

paste("The incorrect number is ", incorrect, ". Replace it with ", replacement, sep = "")


## [1] "The incorrect number is 75. Replace it with 74"
```