

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: The incorrect number is 75 and the correct replacement number is 74. For my solution, I first installed the gmp library so I could use the `factorize()` function to determine the prime factorization of numbers. I then created a matrix mimicking the one provided to us in the problem, and subsequently used the `factorize()` function within a loop to determine the prime factorization of every number in the matrix. I knew that this was a viable method to solving the problem due to the given fact that prime numbers were the key to the problem. After using my loop to determine the prime factorization of every number in the matrix, it instantly became clear that 75 was the odd number out. By running the code `factorize(75)` we can determine that the prime factorization of 75 is $3 \times 5 \times 5$. Of the numbers in our matrix, 75 was the only number that had a prime factorization containing a repeat digit. Every other number had a prime factorization containing two unique digits. I then altered my loop to output any numbers that had a repeat digit in their prime factorization and that is how I solved the first part of the problem since the only output was 75, proving that it was the odd number out. For the second part of the problem, I started by creating a new matrix which included the possible replacement numbers. This matrix contained numbers 70-76 since this was the range we are interested in to find the correct replacement number. I then ran this matrix through a copy of the previous loop, but I reversed the logic so it would output to me any numbers in the matrix without duplicate numbers in their prime factorization. The output from this matrix still contained a few possibilities because, although they didn't have repeat numbers in their prime factorization, they either had 1 or 3 numbers, not 2 like we are looking for in this problem. So, finally I used one final loop to determine which of the remaining couple numbers had a 2 digit prime factorization, since that is what we are interested in for this part of the problem. This final loop outputs 74, which has a prime factorization of 2×37 which can be found by running `factorize(74)`. Since 74's prime factorization contains only 2 numbers which are unique, we know that this is the correct replacement number.

```
library(gmp)

#####Finding Incorrect Number#####
matrix_data <- matrix(c(6, 22, 38, 57, 75, 87,
  10, 26, 39, 58, 77, 91,
  14, 33, 46, 62, 82, 93,
  15, 34, 51, 65, 85, 94,
  21, 35, 55, 69, 86, 95),
  nrow = 5, byrow = TRUE)
```

```

factorization_results <- list()

positions_with_duplicates <- c()

for (i in 1:length(matrix_data)) {
  factors <- factorize(matrix_data[i])

  if (any(duplicated(factors))) {
    positions_with_duplicates <- c(positions_with_duplicates, i)
  }
}
Incorrect_Number = matrix_data[positions_with_duplicates]
Position_Below_Incorrect_Number = positions_with_duplicates - 1
Position_Above_Incorrect_Number = positions_with_duplicates + 1
Lower_Bound = matrix_data[Position_Below_Incorrect_Number] + 1
Upper_Bound = matrix_data[Position_Above_Incorrect_Number] - 1

#####Finding Replacement Number#####

New_Matrix = matrix(Lower_Bound:Upper_Bound)

factorization_results_new <- list()

positions_with_duplicates_new <- c()

for (i in 1:length(New_Matrix)) {
  factors_new <- factorize(New_Matrix[i])

  if (!any(duplicated(factors_new))) {
    positions_with_duplicates_new <- c(positions_with_duplicates_new, i)
  }
}
New_Possible_Numbers = New_Matrix[positions_with_duplicates_new]
for (i in 1:length(New_Possible_Numbers)) {
  Final_Numbers <- factorize(New_Possible_Numbers[i])
  if (length(Final_Numbers) == 2) {
    return(New_Possible_Numbers[i])
  }
}
Replacement_Number = New_Possible_Numbers[i]

#####Printing Answers#####

Incorrect_Number

## [1] 75

Replacement_Number

## [1] 74

```

References

Lucas, A. and Scholz, I. (2024). *gmp: Multiple Precision Arithmetic*. R package version 0.7-5.