

CS:1210 Quiz 4 Version (b)

You have 15 minutes to complete this quiz. Please put away your books, notes, and all electronic devices

1. Suppose that we have already defined a boolean function called `isPrime` that takes as argument a positive integer, say n , and returns `True` if n is a prime and returns `False` otherwise. Now define a boolean function called `isEmptyOfPrimes` that takes two positive integer arguments, say m and n , such that $m \leq n$. The function `isEmptyOfPrimes` should return `True` if *none* of the integers in the range $[m, n]$ are prime; otherwise, it should return `False`. For example, the function call `isEmptyOfPrimes(24, 28)` should return `True`, but `isEmptyOfPrimes(24, 30)` should return `False`. Your definition of `isEmptyOfPrimes` should repeatedly call the function `isPrime`.

Note: You don't have to define `isPrime`, only `isEmptyOfPrimes`.

2. For this problem also, assume that we have already defined a boolean function called `isPrime` (as in Problem 1). Here is a partially completed program that repeatedly prompts the user for a positive integer and outputs all the *prime factors* of that integer. The program repeats this until the user types `done`. The program outputs the factors of each given positive integer in one line. Here is an example interaction between the program and the user. The user enters the positive integers 22, 31, and 64 followed by `done`.

```
Enter a positive integer: 22
Factors: 2 11
Enter a positive integer: 31
Factors: 31
Enter a positive integer: 64
Factors: 2
Enter a positive integer: done
```

The program below has two blanks that need to be filled. Filling one of the blanks involves calling the function `isPrime`.

```
# repeat until user types "done"
while True:
    inputString = input("Enter a positive integer: ")

    # Check if inputString is done and if so break out of loop

    if inputString == "done":
        break

    # This part of the code processes a positive integer
    n = int(inputString)
    factor = 1 # tracks potential factors of n
    # The string variable outputString is used to construct
    # the line of output with all factors of n
    outputString = "Factors: "

    # loop through all potential factors
    while factor <= n:

        # Check if factor is a prime factor of n

        if _____ :

            # Update the outputString

            _____

        factor = factor + 1

    print(outputString)
```