Fitchburg State University

CSC 7014 Practice Computer Programming

Instructor: Nguyen Thai

Due: 10/7/2016 at 5:00 PM

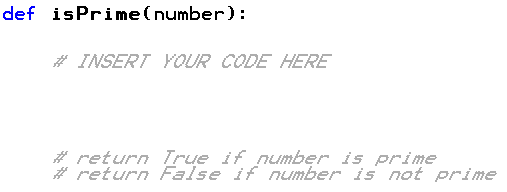
Student:

**CSC 7014 Assignment 4: Prime Number**

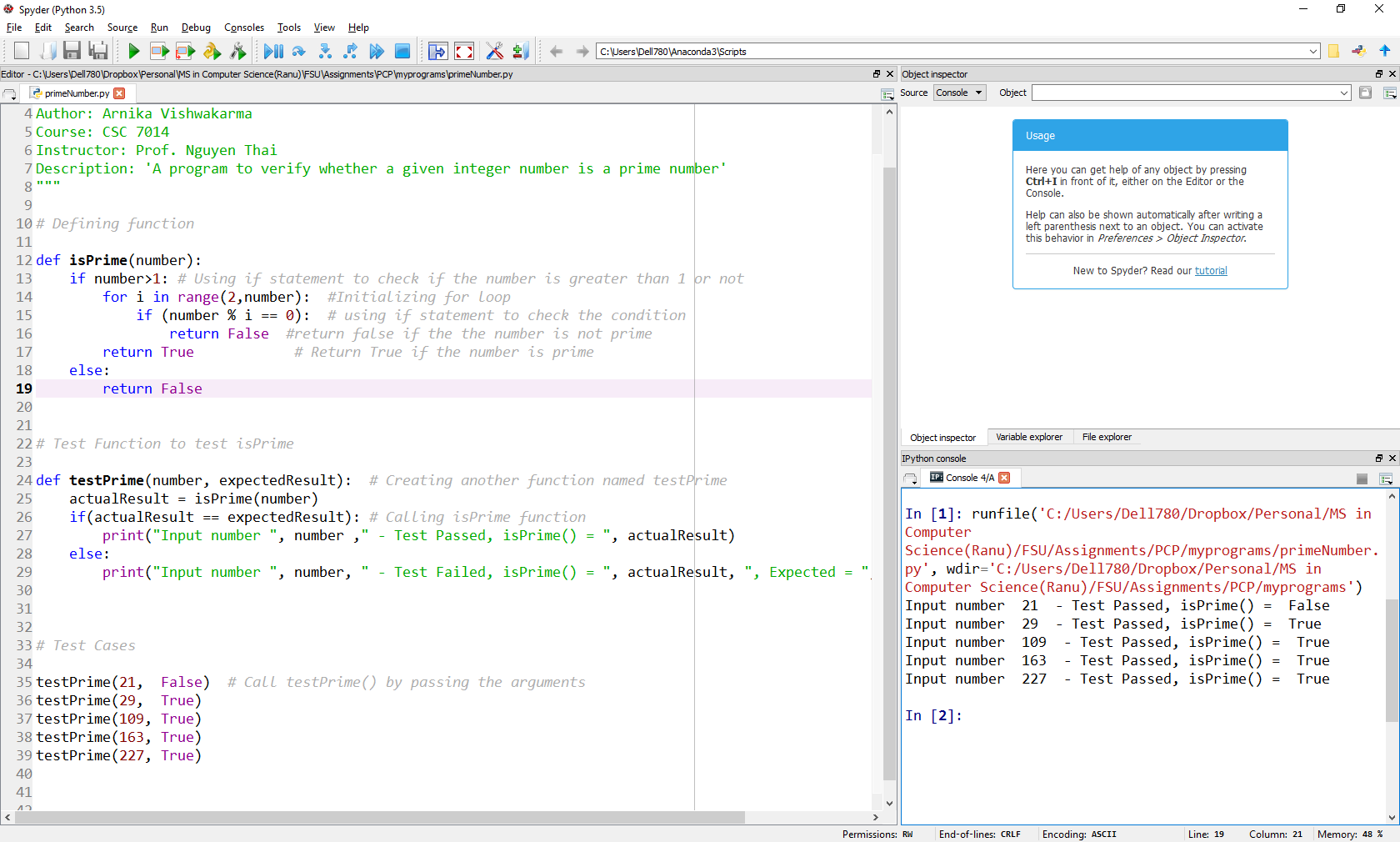
The purpose of this assignment is to learn how to program functions. Your program is to be written in the Python language. You will be graded for output correctness, code comments, code indentation, descriptive variables and source code file header completeness.

As you work through the assignment be sure to answer all questions (type your answers into this document) and take all screenshots as requested (copy them into the document). For the screenshots, you can use the Snipping Tool that is built-in to Windows to capture the important parts of the lab as highlighted in the document below. Do not delete the contents of this file. When finished, you will submit the document source code file and associated data files to the instructor via Blackboard. DO NOT SUBMIT ZIP FILES OR INDIVIDUAL IMAGES. If you have any questions or need any clarification, email the instructor *before* the deadline.

1. In this lab you are to write a program in Python called *primeNumber.py* to verify whether a given integer number is a prime number.
2. An integer greater than 1 is prime if its only positive division is 1 or itself. For example, 2, 3, 5 are prime number, but 4, 6, 8 are not.
3. Design and develop a function called isPrime() with the following signature:



1. Write a test case to test your function, using the numbers: 21, 29, 109, 163 and 227. Print the test results to a console.
2. Before coding, think how you are going to tackle this problem, and write a short description of the logic of your program.
3. **INSERT YOUR DESCRIPTION HERE.**
4. First, we define a function named isPrime() to check whether the number is prime number or not.
5. Inside isPrime() , we used if statement to check whether the input number is greater than 1 or not if it is true then it will go inside the for loop otherwise it will return “False”.
6. We initialized for loop which ranges from (2, number). Inside the for loop we have used if statement to check the condition of the number if it’s a prime number or not.
7. If the (number % i == 0) it will return “False” i.e. it’s not a prime number, because prime numbers are divisible by itself or by 1 otherwise it will return “True”.
8. To test isPrime() we defined another function named **testPrime()** which will take two parameters as number and **expectedResult(The expected Boolean value given by the tester)**, then we call the function **isPrime()** and store the value returned in variable named **actualResult(actual returned Boolean value from isPrime()).**
9. Using if statement we check whether the actualResult is equal to the expectedResult. If both matches the condition, then the test case is passed otherwise it has failed.
10. Finally, we test the cases by calling testPrime() with arguments as number and expected Boolean value.
11. **TAKE A SCREENSHOT** of your input and output, and paste them here. Do not paste your source code in this document.



1. Submit your source code (primeNumber.py) and this document to Blackboard for grading.