PROGRAMMING ASSIGNMENT 3P

Purpose

The purpose of this assignment is to give you practice with repetition using while as well as writing a function in a separate file. No FOR loops are to be used, no break and continue statements allowed. You can use standard math library functions like pow and sqrt (etc.) you cannot use any library functions that might actually compute Fibonacci, Prime or Collatz.

Problem

We want to come up with some common Math computations or sequences:

- Fibonacci sequence is t the infinite sequence 1, 1, 2, 3, 5, 8, 13, … of which the first two terms are 1 and each succeeding term is the sum of the two terms immediately preceding it
- A prime number (or a prime) is a natural number greater than 1 that has no positive divisors other than 1 and itself
- The *Collatz conjecture* which says that no matter what integer you start with, applying this transformation over and over will eventually reach the value 1.

The program will continuously prompt the user for a number (integers only) between 1 and 1000 and do the following for each number input by the user:

a) If the number is less than 1 or greater than 1000 you must display a message indicating that the input is invalid and continue to prompt the user
b) If the number entered is 1000, you should stop the program.
c) In all other cases do the following:
    - Generate the numbers in the Fibonacci series from the beginning until the term in the series is not greater than the number entered. Print each term in the series separated by a tab. Print only 10 numbers on one line. Also print the total number of terms in the sequence. (You MUST use a while loop to generate this sequence).
    - For all numbers in the range 1 to the number entered, calculate all the prime numbers and display each prime number separated by a tab. Print only 10 numbers on one line. Also print the total number of numbers in the sequence. (You MUST use a while loop to generate this sequence).
        - *A prime number is a number greater than 1 and has no divisors other than 1 and the number itself*
        - You must also create a function called is_prime() which takes one input argument (integer) which is the number being checked for prime and returns an integer. The return value is 0 if the number is not prime, and 1 if the number is prime (only 2 possible values to be returned). This function must be in a separate file by itself called prime.c. A function prototype declaration is made at the top of your main source file which needs to be called funwithmath.c (you are being provided starter files).
    - For the Collatz Conjecture, consider this rule for transforming a positive integer **n**:
      If **n** is even, divide it by 2
      If not, multiply it by 3 and add 1
        - For the number that was entered, count the number of transformations it took to reach 1 and also print the sequence of numbers leading to this.
        - (You MUST use a while loop to generate this sequence).

d) In order to compile and run you program with multiple files you need to do the following:

gcc funwithmath.c prime.c -o funwithmath
./funwithmath will execute your program

Output

Output will be sent to standard output (the screen). Since the program is interactive, you must prompt the user for all inputs and you must echo each value that you read from the user. Your answer must include a concise, neat message with the correct value(s) similar to the one found in the sample outputs.

Here is mandatory output needed for comparing your results on Mimir:
(the variable indicate in each printf statement would be the corresponding one that holds the value for that particular operation)
Fibonacci total: printf("Total Fibonacci numbers: %d\n", total_fib);
Prime total: printf("Total Prime numbers: %d\n", total_prime);
Collatz total: printf("Total Collatz transformations: %d\n", total_collatz);

Files to be submitted

You need to submit both files funwithmath.c and prime.c

Compiling your program

gcc funwithmath.c prime.c -lm -o funwithmath
(if you are using math library functions otherwise you can leave out -lm)

Sample Output

```
Please enter a number between 1 and 1000: 88

You entered: 88

Terms upto 88 in the Fibonacci sequence:
1       1       2       3       5       8       13      21      34      55

Total Fibonacci numbers: 10

List of Prime Numbers between 1 and 88:
2       3       5       7       11      13      17      19      23      29
31      37      41      43      47      53      59      61      67      71
73      79      83
Total Prime numbers: 23

Collatz Conjecture for 88:
88->44->22->11->34->17->52->26->13->40->20->10->5->16->8->4->2->1
Total Collatz transformations: 17

Please enter a number between 1 and 1000: 1000

You entered: 1000

Goodbye!
```