**1. Create the Header File (mathfun.h):**

C

#ifndef MATHFUN\_H

#define MATHFUN\_H

int check\_prime(int data);

int test\_digit(int data, int digit); // Accepts digit to check

int test\_ascend(int data);

int test\_descend(int data);

#endif

This header declares the functions John wants to implement in the library, making them accessible to programs that use the library.

**2. Implement the Library Functions (mathfun.c):**

#include "mathfun.h"

#include <stdbool.h> // For boolean data type

bool is\_divisible(int num, int divisor) {

    return num % divisor == 0;

}

int check\_prime(int data) {

    if (data <= 1) return 0; // 1 or less are not prime

    for (int i = 2; i \* i <= data; i++) {

        if (is\_divisible(data, i)) return 0; // Divisible by a number > 1, not prime

    }

    return 1; // Prime number

}

//check for  a particluar digit(eg:3)

int test\_digit(int data, int digit) {

    while (data > 0) {

        if (data % 10 == digit) return 1; // Digit found

        data /= 10;

    }

    return 0; // Digit not found

}

//verifying digits are ascending order

int test\_ascend(int data) {

    int prev\_digit = data % 10;

    data /= 10;

    while (data > 0) {

        int curr\_digit = data % 10;

        if (curr\_digit > prev\_digit) return 0; // Digits not ascending

        prev\_digit = curr\_digit;

        data /= 10;

    }

    return 1; // Digits are ascending

}

//verifying digits are descending order

int test\_descend(int data) {

    int prev\_digit = data % 10;

    data /= 10;

    while (data > 0) {

        int curr\_digit = data % 10;

        if (curr\_digit < prev\_digit) return 0; // Digits not descending

        prev\_digit = curr\_digit;

        data /= 10;

    }

    return 1; // Digits are descending

}

**3. Build the Dynamic Library (libmathfun.so):**

* **Windows (using MinGW):**

Bash

gcc -c -fpic mathfun.c -o mathfun.o

gcc -shared -o libmathfun.so mathfun.o

The -fPIC flag ensures position-independent code (PIC) for dynamic linking. The -shared flag creates a shared library.

**4. Use the Library in Your Program (PrimeNumberWithLibrary.c):**

C

#include <stdio.h>

#include "mathfun.h" // Include the header file

int main() {

    int lower = 100, upper = 1000;

    printf("Output: ");

    for (int num = lower; num <= upper; num++) {

        if (check\_prime(num) && test\_digit(num, 3) && (test\_ascend(num) || test\_descend(num))) {

            printf("%d ", num);

        }

    }

    printf("\n");

    return 0;

}

**5. Link the Library During Compilation:**

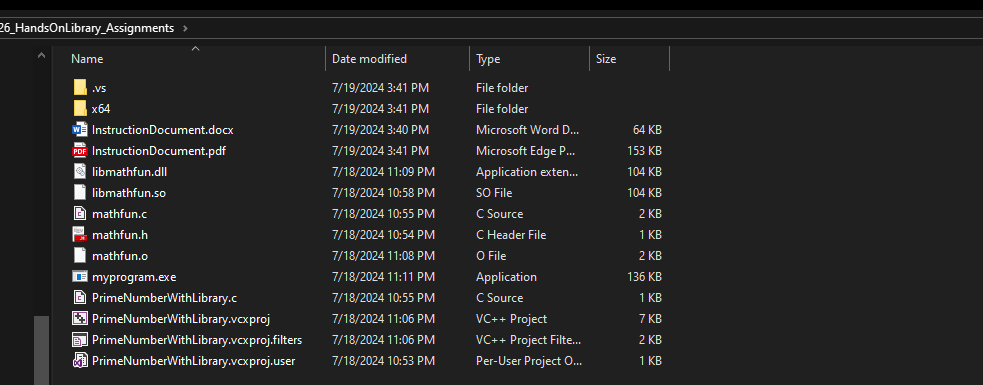
* **Windows (using MinGW):**

Bash

gcc PrimeNumberWithLibrary.c -L. -lmathfun -o PrimeNumberWithLibrary.exe

The -L. specifies the directory containing the library (libmathfun.so or libmathfun.dll). The -lmathfun links against the library name without the .so or .dll extension.

**Source code after completed**



**Running the Program:**

Compile

