Fibonacci Primes

A) A Fibonacci prime is a number in the Fibonacci Sequence that is also prime, meaning that it is only divisible by 1 and itself. The Fibonacci numbers can be defined as: $u\_1 = u\_2 = 1$ and $u\_{n+1} = u\_n + u\_{n-1}$ where $(n > 2)$. The known Fibonacci primes are u_n with n = 3, 4, 5, 7, 11, 13, 17, 23, 29, 43, 47, 83, 131, 137, 359, 431, 433, 449, 509, 569, 571, 2971, 4723, 5387, 9311, 9677, 14431, 25561, 30757, 35999, and 81839. It is thought that there are infinitely many Fibonacci primes, however that theory has not been proven [1]. Another thing known about Fibonacci primes is that for a Fibonacci prime F_n the index n must also be prime (excluding 3), the converse is not true [2].

[1] https://primes.utm.edu/glossary/page.php?sort=FibonacciPrime
[2] https://mathworld.wolfram.com/FibonacciPrime.html

B) Isprime.c

```c
//return 1 if prime, else 0
int isPrime(int val)
{
    int ret=1;
    int i;

    for(i=2; i<=val/2; ++i)
    {
        if(val % i == 0){
            ret=0;
            break;
        }
    }

    if(ret == 0 || val == 1)
        return 0;
    else
        return 1;
}
```

C) main.c

```c
#include "isprime.c"

int main(void)
{
    int iVal=0;
    int result=0;
    printf("Enter a positive integer: ");
    scanf("%d", &iVal);
```

```c
        if(iVal < 1){
                printf("Invalid Input!\n");
                return 0;
        }

        result = isPrime(iVal);

        if(result == 1)
                printf("Result: Prime\n");
        else
                printf("Result: Non-prime\n");

        return EXIT_SUCCESS;
}
```

```
[scole4@id415m40 a1]$ gcc -o test -Wall main.c
[scole4@id415m40 a1]$ ./test
Enter a positive integer: 1
Result: Non-prime
[scole4@id415m40 a1]$ ./test
Enter a positive integer: 2
Result: Prime
[scole4@id415m40 a1]$ ./test
Enter a positive integer: 8
Result: Non-prime
[scole4@id415m40 a1]$ ./test
Enter a positive integer: 17
Result: Prime
```

D) isfib.c

```c
//return 1 if yes, else 0
int isFib(int val)
{
        int a=0;
        int b=1;

        if(val==a || val==b)
                return 1;

        int nextNum = 1;
        while(nextNum<=val)
        {
                if(val==nextNum)
```

```c
                    return 1;
            a=b;
            b=nextNum;
            nextNum=a+b;
        }

        return 0;
    }
```

E) fibtest.c

```c
int main(void)
{
    int iVal;
    int fibRes=0;
    printf("Enter an integer >= 0: ");
    scanf("%d", &iVal);

    fibRes = isFib(iVal);

    if(fibRes == 1)
            printf("Result: Fibonacci Number\n");
    else
            printf("Result: Not a Fibonacci Number\n");


    return EXIT_SUCCESS;
}
```

```
[scole4@id415m40 a1]$ gcc -o fibtest -Wall fibtest.c
[scole4@id415m40 a1]$ ./fibtest
Enter an integer >= 0: 0
Result: Fibonacci Number
[scole4@id415m40 a1]$ ./fibtest
Enter an integer >= 0: 1
Result: Fibonacci Number
[scole4@id415m40 a1]$ ./fibtest
Enter an integer >= 0: 2
Result: Fibonacci Number
[scole4@id415m40 a1]$ ./fibtest
Enter an integer >= 0: 3
Result: Fibonacci Number
[scole4@id415m40 a1]$ ./fibtest
Enter an integer >= 0: 4
Result: Not a Fibonacci Number
[scole4@id415m40 a1]$ ./fibtest
Enter an integer >= 0: 5
Result: Fibonacci Number
[scole4@id415m40 a1]$ ./fibtest
Enter an integer >= 0: 6
Result: Not a Fibonacci Number
[scole4@id415m40 a1]$ ./fibtest
Enter an integer >= 0: 7
Result: Not a Fibonacci Number
```

F) findprimefibs.c

```c
void findPrimeFibs(int min, int max)
{
    int count = min;
    while(count <= max)
    {
        if(isFib(count) == 1 && isPrime(count) == 1)
            printf("%d\n", count);
        count++;
    }
}
```

```
int main(void)
{
    int val1;
    int val2;

    printf("Enter the smallest number of your range: ");
    scanf("%d", &val1);

    printf("Enter the largest number of your range: ");
    scanf("%d", &val2);

    findPrimeFibs(val1,val2);

    return EXIT_SUCCESS;
}
```

10 to 100

```
[scole4@id415m40 a1]$ ./primefibs
Enter the smallest number of your range: 10
Enter the largest number of your range: 100
13
89
```

1597 to 1597

```
[scole4@id415m40 a1]$ ./primefibs
Enter the smallest number of your range: 1597
Enter the largest number of your range: 1597
1597
```

Directory Listing

```
[scole4@id415m40 a1]$ ls
fibtest  fibtest.c  findprimefibs.c  isfib.c  isprime.c  main.c
```