



CS 2263 - FR01A

Assignment 1

By Ngoc Phuong Anh Nguyen - 3712361

September 2021

Questions:

Question A: Describe the Fibonacci primes: define what they are, describe what's known about them, include at least TWO references:

Fibonacci Primes are prime integers that in the Fibonacci sequence, which means a number in the list must be larger than 1 and only can be divided by itself and 1. The smallest Fibonacci prime number is 2. The largest known Fibonacci prime number is $F_{3340367}$, which has 698096 digits long.

References:

- ▶ [Fibonacci number - Wikipedia](#)
- ▶ [Fibonacci prime - Wikipedia](#)
- ▶ [Largest known Fibonacci prime | Googology Wiki | Fandom \(wikia.org\)](#)

Question B: Write a C function (show the source code) that accepts as one parameter: an integer, and determines if it is a prime number. Return 0 if it is not and 1 if it is.

```
#include <stdlib.h>
int isPrime(int i)
{
    if(i == 0 || i == 1 || i < 0)
    {
        return 0;
    }
    else if(i == 2)
    {
        return 1;
    }
    else
    {
        int j;
        for (j = 2; j < i^(1/2); j++)
        {
            if(i%j == 0)
            {
                return 0;
            }
        }
        return 1;
    }
}
```

Figure 1: Source Code of isprime.c

Question C: Write a C program (show the source code and the screen capture for the test run) that prompts for an integer value and prints a message if the number is a prime. The test program should use isPrime function stored in a separate file. The compilation command may look like this:

```
$gcc testingprimes.c isprime.c -o prog1
```

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  int isPrime(int i);
4
5  int main(int argc, char ** argv)
6  {
7      int value;
8      printf("Input the value you want to check: ");
9      int check = scanf("%d", &value);
10     if(check != 1)
11     {
12         printf("Unable to read the value\n");
13         return EXIT_FAILURE;
14     }
15     if(isPrime(value) == 1)
16     {
17         printf("%d is a prime\n", value);
18     }
19     else
20     {
21         printf("%d is not a prime\n", value);
22     }
23     return EXIT_SUCCESS;
24 }
```

Figure 2: The Source Code of testingprimes.c

```
[anguyen5@gc112m30 Assignment 1]$ gcc -c isprime.c
[anguyen5@gc112m30 Assignment 1]$ gcc -c testingprimes.c
[anguyen5@gc112m30 Assignment 1]$ gcc testingprimes.c isprime.c -o prog1
[anguyen5@gc112m30 Assignment 1]$ ./prog1
Input the value you want to check: 5
5 is a prime
[anguyen5@gc112m30 Assignment 1]$ ./prog1
Input the value you want to check: 4
4 is not a prime
[anguyen5@gc112m30 Assignment 1]$ ./prog1
Input the value you want to check: 2
2 is a prime
[anguyen5@gc112m30 Assignment 1]$ ./prog1
Input the value you want to check: -1
-1 is not a prime
```

Figure 3: The screen capture for the test run

Question D: Write a C function (show the source code) that accepts an integer as a parameter and determines if it is a number from the Fibonacci sequence. Return 0 if it is not and 1 if it is.

```
1  #include <stdlib.h>
2  int isFib(int i)
3  {
4      if(i == 0 || i == 1)
5      {
6          return 1;
7      }
8      else if(i < 0)
9      {
10         return 0;
11     }
12     else
13     {
14         int a = 0;
15         int b = 1;
16         int c = a + b;
17         while(c <= i)
18         {
19             if(i == c)
20             {
21                 return 1;
22             }
23             a = b;
24             b = c;
25             c = a + b;
26         }
27         return 0;
28     }
29 }
```

Figure 4: The Source Code of isfib.c

Question E: Write a C program (show the source code and the screen capture for the test run) that prompts for an integer value and prints a message if the number is from the Fibonacci sequence. The test program should use isFib function stored in a separate file. The compilation command may look like this:

```
$gcc testingfibs.c isfib.c -o prog2
```

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 int isFib(int i);
4
5 int main(int argc, char ** argv)
6 {
7     int value;
8     printf("Input the value you want to check: ");
9     int check = scanf("%d", &value);
10    if(check != 1)
11    {
12        printf("Unable to read the value\n");
13        return EXIT_FAILURE;
14    }
15    if(isFib(value) == 1)
16    {
17        printf("%d is a Fibonacci\n", value);
18    }
19    else
20    {
21        printf("%d is not a Fibonacci\n", value);
22    }
23    return EXIT_SUCCESS;
24 }
```

Figure 5: The Source Code of testingfibs.c

```
[anguyen5@gc112m30 Assignment 1]$ gcc -c isfib.c
[anguyen5@gc112m30 Assignment 1]$ gcc -c testingfibs.c
[anguyen5@gc112m30 Assignment 1]$ gcc testingfibs.c isfib.c -o prog2
[anguyen5@gc112m30 Assignment 1]$ ./prog2
Input the value you want to check: 0
0 is a Fibonacci
[anguyen5@gc112m30 Assignment 1]$ ./prog2
Input the value you want to check: -1
-1 is not a Fibonacci
[anguyen5@gc112m30 Assignment 1]$ 5
5: Command not found.
[anguyen5@gc112m30 Assignment 1]$ ./prog2
Input the value you want to check: 5
5 is a Fibonacci
[anguyen5@gc112m30 Assignment 1]$
```

Figure 6: The screen capture for the test run

Question F: Now, write a C program (show the source code and the screen captures for the test runs) that accepts two integer values x1 and x2 specifying the range of integers (all inclusive), and tests all the values in the specified range printing only the Fibonacci primes in this range. Use the functions defined in separate files isprime.c and isfib.c. Run the program on each of the ranges:

i. 10 to 100

ii. 1597 to 1597

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  int isPrime(int i);
4  int isFib(int i);
5
6  int main(int argc, char **argv)
7  {
8      int x1;
9      int x2;
10     int i;
11     printf("Input the beginning of the range: ");
12     int check1 = scanf("%d", &x1);
13
14     printf("Input the ending of the range: ");
15     int check2 = scanf("%d", &x2);
16
17     if(check1 != 1 || check2 != 1)
18     {
19         printf("Unable to read the value\n");
20         return EXIT_FAILURE;
21     }
22     printf("List of Fibonacci Primes: ");
23     for(i = x1; i <= x2; i++)
24     {
25         if(isPrime(i) == 1 && isFib(i) == 1)
26         {
27             printf("%d\t", i);
28         }
29     }printf("\n");
30     return EXIT_SUCCESS;
31 }
```

Figure 7: The Source Code of main.c


```
[anguyen5@gcl12m30 Assignment 1]$ gcc -c main.c
[anguyen5@gcl12m30 Assignment 1]$ gcc main.c isprime.c isfib.c -o prog3
[anguyen5@gcl12m30 Assignment 1]$ ./prog3
Input the beginning of the range: 10
Input the ending of the range: 100
List of Fibonacci Primes: 13    89
[anguyen5@gcl12m30 Assignment 1]$ ./prog3
Input the beginning of the range: 1597
Input the ending of the range: 1597
List of Fibonacci Primes: 1597
[anguyen5@gcl12m30 Assignment 1]$ █
```

Figure 8: The screen capture for the test run

```
[anguyen5@gcl12m30 Assignment 1]$ dir
isfib.c    isprime.o  prog1    testingfibs.c    testingprimes.o
isfib.o    main.c     prog2    testingfibs.o
isprime.c  main.o     prog3    testingprimes.c
[anguyen5@gcl12m30 Assignment 1]$ █
```

Figure 9: List of Files in the Directory