Assignment 7

Iterative Control

```c
#include <stdio.h>


int main() {
    int choice, N, num1, num2, num, i, j, isPrime, isArmstrong, a, b, temp, nextPrime;

    int fib[20];
    // Initialize Fibonacci series
    fib[0] = 0;
    fib[1] = 1;
    for (i = 2; i < 20; i++) {
        fib[i] = fib[i - 1] + fib[i - 2];
    }

    // 1: Find the Nth term of the Fibonacci series
    printf("Enter N to find the Nth term: ");
    scanf("%d", &N);
    printf("The Nth term of the Fibonacci series: %lld\n\n", fib[N]);

    // 2: Print the first N terms of the Fibonacci series
    printf("Enter N to print the first N terms: ");
    scanf("%d", &N);
    printf("The first %d terms of the Fibonacci series:\n", N);
    for (i = 0; i < N; i++) {
        printf("%lld ", fib[i]);
    }
```

```
printf("\n\n");
// 3: Check if a number is in the Fibonacci series
printf("Enter a number to check if it's in the Fibonacci series: ");
scanf("%d", &num);
int isFibonacci = 0;
for (i = 0; i < 50; i++) {
    if (num == fib[i]) {
        isFibonacci = 1;
        break;
    }
}
if (isFibonacci) {
    printf("%d is in the Fibonacci series.\n\n",num);
} else {
    printf("%d is not in the Fibonacci series.\n\n",num);
}

// 4: Calculate the HCF of two numbers
printf("Enter two numbers to calculate their HCF: ");
scanf("%d %d", &num1, &num2);
a = num1;
b = num2;
while (b != 0) {
    temp = b;
    b = a % b;
    a = temp;
}
printf("HCF of %d and %d: %d\n\n", num1, num2, a);
```

```c
// 5: Check if two numbers are co-prime
printf("Enter two numbers to check if they are co-prime: ");
scanf("%d %d", &num1, &num2);
a = num1;
b = num2;
while (b != 0) {
    temp = b;
    b = a % b;
    a = temp;
}
if (a == 1) {
    printf("%d and %d are co-prime.\n\n", num1, num2);
} else {
    printf("%d and %d are not co-prime.\n\n", num1, num2);
}

// 6: Print all prime numbers under 100
printf("Prime numbers under 100:\n");
for (i = 2; i < 100; i++) {
    isPrime = 1;
    for (j = 2; j < i; j++) {
        if (i % j == 0) {
            isPrime = 0;
            break;
        }
    }
```

```
    if (isPrime) {
        printf("%d ", i);}
}
printf("\n\n");
// 7: Print prime numbers between two numbers
printf("Enter two numbers to print prime numbers between them: ");
scanf("%d %d", &num1, &num2);
printf("Prime numbers between %d and %d:\n", num1, num2);
for (i = num1; i <= num2; i++) {
    if (i < 2) {
        continue;
    }
    isPrime = 1;
    for (j = 2; j * j <= i; j++) {
        if (i % j == 0) {
            isPrime = 0;
            break;
        }
    }
    if (isPrime) {
        printf("%d ", i);
    }
}
printf("\n\n");

// 8: Find the next prime number after a given number
printf("Enter a number to find the next prime number after it: ");
scanf("%d", &num);
nextPrime = num + 1;
```

```
while (1) {
    isPrime = 1;
    for (i = 2; i * i <= nextPrime; i++) {
        if (nextPrime % i == 0) {
            isPrime = 0;
            break;
        }
    }
    if (isPrime) {
        printf("Next prime number after %d: %d\n\n", num, nextPrime);
        break;
    }
    nextPrime++;
}

// 9: Check if a number is an Armstrong number
printf("Enter a number to check if it's an Armstrong number: ");
scanf("%d", &num);
isArmstrong = 0;
int sum = 0;
int originalNum = num;
while (originalNum != 0) {
    int digit = originalNum % 10;
    sum += digit * digit * digit;
    originalNum /= 10;
}
if (num == sum) {
    isArmstrong = 1;
}
```

```c
    if (isArmstrong) {
        printf("%d is an Armstrong number.\n\n", num);
    } else {
        printf("%d is not an Armstrong number.\n\n", num);
    }

    // 10: Print all Armstrong numbers under 1000
    printf("Armstrong numbers under 1000:\n");
    for (i = 1; i < 1000; i++) {
        isArmstrong = 0;
        int sum = 0;
        int originalNum = i;
        while (originalNum != 0) {
            int digit = originalNum % 10;
            sum += digit * digit * digit;
            originalNum /= 10;
        }
        if (i == sum) {
            isArmstrong = 1;
        }
        if (isArmstrong) {
            printf("%d ", i);
        }
    }
    printf("\n");

    return 0;
}
```

```
Enter N to find the Nth term: 10
The Nth term of the Fibonacci series: 55

Enter N to print the first N terms: 10
The first 10 terms of the Fibonacci series:
0 1 1 2 3 5 8 13 21 34

Enter a number to check if it's in the Fibonacci series: 23
23 is not in the Fibonacci series.

Enter two numbers to calculate their HCF: 12
50
HCF of 12 and 50: 2

Enter two numbers to check if they are co-prime: 56
48
56 and 48 are not co-prime.

Prime numbers under 100:
2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97

Enter two numbers to print prime numbers between them: 12
24
Prime numbers between 12 and 24:
13 17 19 23

Enter a number to find the next prime number after it: 23
Next prime number after 23: 29

Enter a number to check if it's an Armstrong number: 56
56 is not an Armstrong number.

Armstrong numbers under 1000:
1 153 370 371 407
```