

DAA Lab – Assignment 2

Divide and Conquer Algorithms

1. [Collatz Conjecture] Consider the following algorithm to generate a sequence of numbers. Start with an integer n . If n is even, divide by 2. If n is odd, multiply by 3 and add 1. Repeat this process with the new value of n , terminating when $n = 1$. For example, the following sequence of numbers will be generated for $n = 22$: 22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1. Count the sequence length.

Code:

```
def collatz (num):  
    print(num,end=" ")  
    if (num==1):  
        print()  
        return  
    if (num%2==0):  
        return collatz(num/2)  
    else:  
        return collatz(3*num+1)  
  
n = int(input("Enter number: "))  
collatz(n);
```

Output:

```
~/DAA-Exercise2$ python3 collatz.py  
Enter number: 22  
22 11.0 34.0 17.0 52.0 26.0 13.0 40.0 20.0 10.0 5.0 16.0 8.0 4.0 2.0 1.0  
~/DAA-Exercise2$ python3 collatz.py  
Enter number: 19  
19 58 29.0 88.0 44.0 22.0 11.0 34.0 17.0 52.0 26.0 13.0 40.0 20.0 10.0 5.0 16.0 8.0 4.0 2.0 1.0  
~/DAA-Exercise2$ python3 collatz.py  
Enter number: 96  
96 48.0 24.0 12.0 6.0 3.0 10.0 5.0 16.0 8.0 4.0 2.0 1.0  
~/DAA-Exercise2$ python3 collatz.py  
Enter number: 51  
51 154 77.0 232.0 116.0 58.0 29.0 88.0 44.0 22.0 11.0 34.0 17.0 52.0 26.0 13.0 40.0 20.0 10.0 5.0 16.0 8.0 4.0 2.0 1.0
```

2. Implement Fibonacci Series using Iterative algorithm, Recursive algorithm and Golden Ratio.

Code:

Iterative

```
def fib_it(n):
    a,b = 0,1;
    print(0,1,end=" ")
    for i in range(3,n+1):
        c = a+b
        print(c,end=" ")
        a,b = b,c
    print()

num = int(input("Enter number: "))
fib_it(num)
```

Recursive

```
def fib_rec(n):
    if (n==1):
        return 0;
    if (n==2):
        return 1;
    else:
        return fib_rec(n-1)+fib_rec(n-2);

num = int(input("Enter number: "))
for i in range(1,num+1):
    print(fib_rec(i),end=" ")
print()
```

Golden ratio

```
def fib_rec(n):
    if (n==1):
        return 0;
    if (n==2):
        return 1;
    else:
        return fib_rec(n-1)+fib_rec(n-2);

num = int(input("Enter number: "))
for i in range(1,num+1):
    print(fib_rec(i),end=" ")
print()
```

Output:

```
~/DAA-Exercise2$ python3 fib_it.py
Enter number: 10
0 1 1 2 3 5 8 13 21 34
~/DAA-Exercise2$ python3 fib_rec.py
Enter number: 10
0 1 1 2 3 5 8 13 21 34
~/DAA-Exercise2$ python3 fib_gr.py
Enter number: 10
0 1 1 2 3 5 8 13 21 34
~/DAA-Exercise2$
~/DAA-Exercise2$ python3 fib_it.py
Enter number: 16
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377 610
~/DAA-Exercise2$ python3 fib_rec.py
Enter number: 16
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377 610
~/DAA-Exercise2$ python3 fib_gr.py
Enter number: 16
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377 610
~/DAA-Exercise2$
```

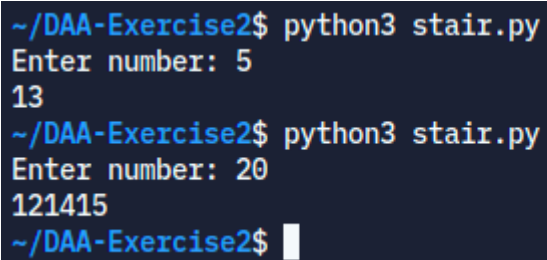
3. Count ways to reach the nth stair using step 1, 2 or 3.

Code:

```
def findStep(n):
    if ( n == 0 ):
        return 1
    elif (n < 0):
        return 0
    else:
        return findStep(n - 3) + findStep(n - 2) + findStep(n - 1)

# Driver code
num = int(input("Enter number: "))
print(findStep(num))
```

Output:



```
~/DAA-Exercise2$ python3 stair.py
Enter number: 5
13
~/DAA-Exercise2$ python3 stair.py
Enter number: 20
121415
~/DAA-Exercise2$
```

4. Karatsuba algorithm for fast multiplication using Divide and Conquer algorithm.

Code:

```
def karatsuba(x,y):
    if len(str(x)) == 1 or len(str(y)) == 1:
        return x*y

    else:
        n = max(len(str(x)),len(str(y)))
        nby2 = n // 2

        a = x // 10**(nby2)
        b = x % 10**(nby2)
        c = y // 10**(nby2)
        d = y % 10**(nby2)

        ac = karatsuba(a,c)
        bd = karatsuba(b,d)
        ad_plus_bc = karatsuba(a+b,c+d) - ac - bd

        #writing n as 2*nby2 takes care of both even and odd n
        prod = ac * 10**(2*nby2) + (ad_plus_bc * 10**nby2) + bd

    return prod

num1 = int(input("Enter first number: "))
num2 = int(input("Enter second number: "))
print("Product:",karatsuba(num1,num2))
```

Output:

```
~/DAA-Exercise2$ python3 kara.py
Enter first number: 20
Enter second number: 12
Product: 240
~/DAA-Exercise2$ python3 kara.py
Enter first number: 561
Enter second number: 905
Product: 507705
~/DAA-Exercise2$
```