

# Bangladesh University of Business and Technology (BUBT)



## Lab Report

**Course code:** CSE 352

**Course title:** Artificial Intelligence

**Experiment no:** 02

**Experiment name:** Write a Python program to print the Fibonacci series up to n terms and also find whether the number is a strong number or not. Use the function to print the pattern number.

### Submitted by:

Name: Tasnuba Khanom  
Id: 19201103106  
Intake: 43  
Section: 2  
Program: B.Sc. Engg in CSE  
Semester: Spring-2022

### Submitted to:

Humayra Ahmed  
Lecturer, Department of CSE  
Bangladesh University of Business and Technology  
(BUBT)

**Date of Submission:** 21 / 09 / 2022

Report no: 02

Report name: Write a Python program to print the Fibonacci series up to n terms and also find whether the number is a strong number or not. Use the function to print the pattern number.

Code Input/Output:

For Fibonacci series:

```
n = int(input("Number of series:"))
a, b = 0, 1
count = 0
if n <= 0:
    print("Invalid input!!Please enter a positive integer")
else:
    print("Fibonacci sequence is:")
    while count < n:
        print(a)
        temp = a + b
        a = b
        b = temp
        count += 1
```

```
Number of series:5
Fibonacci sequence is:
0
1
1
2
3
```

For strong number:

```
num = int(input("Enter a number: "))
sum = 0
temp = num
while(temp > 0):
    fact = 1
    r = temp % 10
    for i in range(1, r + 1):
        fact = fact * i
    sum = sum + fact
    temp = temp // 10
if(sum==num):
    print("Yes", num , "is strong number")
else:
    print("No" , num, "is not a strong number")
```

```
Enter a number: 145
Yes 145 is strong number
```

**For number pattern:**

```
def function(r,c):  
    for i in range(1,r+1):  
        for j in range(i,i+c):  
            print(j,end=" ")  
        print("\n")  
  
r=int(input("Enter row:"))  
c=int(input("Enter column:"))  
function(r,c)
```

```
1 2 3 4 5  
2 3 4 5 6  
3 4 5 6 7  
4 5 6 7 8  
5 6 7 8 9
```

**Description:**

**Algorithm:**

**For fibonacci series:**

- Taking a integer type "n" input that represents number of series.
- Suppose we have three integer types "a," "b," and "count" with initial values of zero.
- If "n" is smaller than or equal to 0, then it will print invalid input.
- Or else it will print the Fibonacci series with the initials,
- We have to look if count is smaller than n. If it is true, then print the first value.
- We will use a temp variable which will work for the second and n-term series.
- At first, we will add a and b. After that, we will swap and print them.

**For strong number:**

- Take an integer type input "num", which represents "Enter a number".
- Now switch the value of num in temp with that of a variable whose initial value is 0.
- Using a while loop, if temp is greater than 0, then the fact variable will become 1 and variable r will have temp modulus.
- After getting r, we will use it for a loop whose conditions are (1, r+1).
- If the value is in the range, the fact variable will be upgraded by "fact\*i".
- The sum will be the addition of the previous sum and the fact. temp will be upgraded by floor division.
- Afterwards, print "it is a strong number" if sum equals num.
- Else print "it is not strong number".

**For number pattern:**

- We will take r and c for input row size and column size respectively.
- Take a function whose parameters are "r" and "c".
- If i equals 1 and r+1 then there will be a work of for loop.
- similarly, we will use another for loop for c using j.
- After going through the nested for loop it will print the number of row and column in the pattern.
- Then call the function for getting the above process.

### Used functions:

- Input(): The input() function reads a line from the information (often from the user), removes the terminating newline to turn the line into a string, and then returns the string.

Syntax: input(prompt)

- int(): The int() method in Python creates an integer number from the provided value. If no arguments are provided, the int() function will return 0. It will also generate an integer object from a number or string, let's say x.

Syntax: int(value, base)

value = A number or a string that can be converted into an integer number

base = A number representing the number format. Default value 10

- If else(): The true and false parts of a given condition are both executed using the if-else() expression. If the condition is true, the code in the if block is run, and if it is false, the code in the else block is run.

Syntax of if...else

if test expression:

Body of if

else:

Body of else

- for loop(): When you wish to repeat a section of code a certain number of times, you use for loops. The for statement in Python executes the block each time it iterates over the elements of a sequence in order. Compare the "while" loop, which is used when a condition needs to be verified after each iteration, to the "for" statement.

Here is the basic structure of a for loop in Python:

for [item] in [sequence]:

Run code

- while loop(): A while loop repeats a block of code an unknown number of times until a condition is no longer met. A while loop will always first check the condition before running. If the condition evaluates to True, then the loop will continue to run the code while the condition remains True.

Syntax of while Loop in Python

while expression:

statement(s)

- def(): The def keyword in Python is used to define a function; it is prefixed with a user-supplied function name to construct a user-defined function. A function in Python is a logical unit of code that includes a series of statements that are indented and are given names using the "def" keyword. The most popular keyword in Python is "def."

Syntax:

def function name:

function definition statements..

- Print(): The print() method outputs the text to the normal external device, such as the display. Any object, including a string, can serve as the text. Before being displayed on the screens, the item will be changed into a string.

Syntax: print(message)