

**ENGR 101 – Introduction to Programming**  
**Study Questions – Week 7**

Q1 – Write a recursive function that takes an input  $n$  and sums all the numbers till zero.

(e.g.: if the input is 5, the output should be 15 summing  $(5 + 4 + 3 + 2 + 1)$  ).

Q2 – Write a recursive function that takes an even number as an input, then adds all even numbers starting from that number until 0. (e.g.: if the input is 6, the output should be 12 summing  $(6 + 4 + 2)$ ).

Q3 – in the last question we assumed that that input will be an even number, now redefine the function so it can take an odd number and still sum all even numbers until 0.

(e.g.: if the input is 7, the output should be 12 summing  $(6 + 4 + 2)$ ).

Q4 – The following recursive function is supposed to add all negative numbers from a given number until 0. (e.g.: if the input is -3, then the output should be -6), but our base in the function is not the correct. Can you change the base in the following function so it works as intended?

```
def add_negatives(n):  
    if n == 1:  
        return 1  
    return n + add_negatives(n+1)  
print(add_negatives(-3))
```

**ENGR 101 – Introduction to Programming**  
**Study Questions – Week 7**

Q5 – Write a recursive function that takes a number consisting of at least one digit as an input and outputs the number of the digits of that number. (e.g.: if the input is 123, then it should return 3, number of digits in the given number. You can NOT use `len()`).

Q6 – Solve the previous question with both for and while loops each in a separate function that will return the number of digits. You may use `len()` here.

Q7 - Think of a recursion version of the function  $f(n) = 3*n$ , i.e., the multiples of 3. Write a recursive Python function that will compute  $f(n)$  for a given number  $n$ . Now, challenge yourself to write the same function in a non-recursive way using loop(s).

Q8 - Write a recursive function `power(x,y)` to calculate the value of  $x$  raised to the power of  $y$ . Now, challenge yourself to write the same function in a non-recursive way using loop(s).

Q9 - Write a recursive function `sum_of_digits()` that takes an integer and returns the sum of the digits in the integer. Now, challenge yourself to write the same function in a non-recursive way using loop(s).

Q10 - Convert the while loop from the following code to a recursive function.

```
def check(i,t):  
    while i<=100:  
        if i%2==0:  
            t+=i  
        i+=1  
    return t
```

Q11 - The sum of first 100 positive integers is 5050. You can check this using the  $\frac{n(n+1)}{2}$  formula (bonus: you can also read [this story](#)). We wrote a recursive function to compute the above-mentioned sum but it does not work correctly. Find out the errors in the code below and correct them. In the second step, challenge yourself to write the same function in a non-recursive way using loop(s).

```
def recur_sum(start, max):  
    if start>max:  
        return max  
    else:  
        return 1 + recur_sum(start+1, max)  
  
print recur_sum(1, 100)|
```

Q12- Write a function, which counts and prints from given number till 0. Whenever it reaches number 20 it assigns integer 100 to counting variable.

Example: 50,49,48...22,21,100,99,98...22,21,100...

Q13- Fibonacci Numbers are as follows: 0 1 1 2 3 5 8 13 21...

Read more about Fibonacci recursive function:

<https://medium.com/@johanna.fulghum/write-the-fibonacci-sequence-in-every-computational-complexity-9adf5ef12775>

By definition, the first two numbers in the Fibonacci sequence are 1 and 1, or 0 and 1, depending on the chosen starting point of the sequence, and each subsequent number is the sum of the previous two.

Write a fibonacci(n) function which takes a number as parameter and returns the *n*<sup>th</sup> number in the sequence (consider 0 as the starting point!) by using;

**ENGR 101 – Introduction to Programming**  
**Study Questions – Week 7**

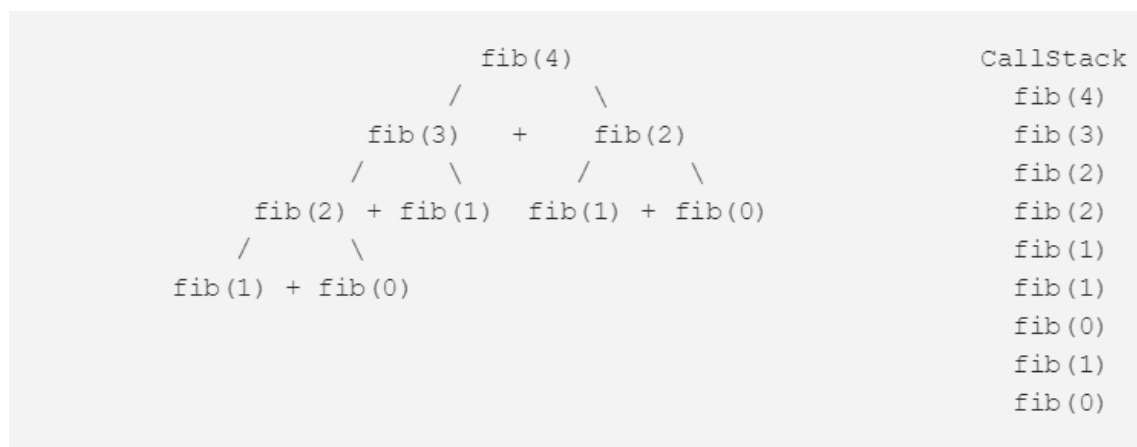
- a)for loop
- b)while loop
- c)recursive

e.g. seq. number: 0 1 1 2 3 5 8 13 21 ...

Num. order: 0 1 2 3 4 5 6 7 8 ...

The Num. order will be input to fibonacci(n) function.

e.g. When Fibonacci is called with Fibonacci(4)



Q14 – The following recursive function is supposed to multiply from a given number  $n$  until 0. (e.g.: if the output is 3, then the output should be 6 multiplying  $3 * 2 * 1$ ). Yet, it's not working as intended. Can you fix it?

```
def multiply_from_n_to_zero(n):
    if n == 0:
        return 0
    return n * multiply_from_n_to_zero(n-1)
```

**ENGR 101 – Introduction to Programming**  
**Study Questions – Week 7**

Q15 – Write a function that takes a number  $n$  then compares it with a random integer number between (0 and  $n**2$ ) until the random number equals the given number. The function should print all random numbers each time it compares it with the given number. When they are equal, 'True' Should be printed and the while loop should be stopped. Use While loop!

Q16 - You wrote a *countdown()* function with both for and while loop in the previous weeks. Now modify your countdown function and it should take two numbers as an input. It should terminate the code when starts to count down from the first number and reaches to the second number. You program should not print the second number.

For example:

countdown(10,7) --- The output will be

10

9

8

Q17 - Now, modify the countdown function in question 9 with taking inputs from the keyboard. It should check;

- If first number equal or smaller than second one, it should give you warnings and enable you enter new inputs until first number is greater than the second one.
- If not, it should print all numbers from first input to the second input and terminate the program.

For example:

countdown()

>> Enter your first number:

>> Enter your second number:

and result should be printed here.

Q18 – Write a program that will ask the user repeatedly to enter a number, until the entered number equals a random integer from 1 to 3. If the user enters a false guess the program should ask the user to enter a different number. If the two numbers are equal, the program should terminate.

Q19 – What is the output of the following piece of code? Justify your answer.

```
while True:
    rand_num = random.randrange(0, 10, 2)
    if rand_num%2==0:
        print(True)
    else:
        print(False)
        break
```

Q20 – Write a program that chooses a letter randomly from the following letters 'abc' and accordingly it will print that chosen letter. (e.g.: if the random letter is 'a', your program should print the following: The letter is a.