NAME: ABDULKAREEM MARYAM TOLOHUNTOMILADE
COURSE: BACK-END DEVELOPMENT
ASSIGNMENT:1

QUESTIONS

1. Write an algorithm to add two numbers.
2. Write an algorithm to find the factorial of a number.
3. Write an algorithm to find the Fibonacci sequence
4. Write an algorithm to sort a given set of numbers. (Bubble sort)
5. Write an algorithm to convert a given number from one base to another.

SOLUTIONS

SOLUTION 1:

```
a) Start
b) Input an integer number A
c) If non-integer number is entered by user, ask user to repeat input.
d) Input an integer number B
e) If non-integer number is entered by user, ask user to repeat input.
f) Define an output integer Sum to take the sum of the two numbers.
g) Sum = A + B
h) Print Sum
i) Stop
```

SOLUTION 2:

Factorial, in mathematics, the product of all positive integers less than or equal to a given positive integer and denoted by that integer and an exclamation point. Thus, factorial seven is written 7!, meaning 1 × 2 × 3 × 4 × 5 × 6 × 7.

Hence below is an algorithm to find factorial of any given positive integer:

```
a) Start
b) Enter a positive integer number Num, ask user to repeat entry for wrong input.
c) Declare an integer variable – Factorial and give it an initial value of 1.
d) Check that user entered integer number and prompt them to try again if not.
e) Initiate a loop
    a. While Num is greater than zero
        i. Multiply Factorial by Num and store result in Factorial
        ii. Subtract 1 from Num
    b. End While Loop
f) Print out Factorial
g) Stop
```

SOLUTION 3:

The Fibonacci sequence is a set of numbers that starts with a one or a zero, followed by a one, and proceeds based on the rule that each number (called a Fibonacci number) is equal to the sum of the preceding two numbers.

a) Start
b) Enter and read a positive integer for the range of the numbers **Range**
c) Ask user to repeat entry for wrong input.
d) Declaring integer variables, Let;
    i. FirstNumber=0
    ii. SecondNumber=1
    iii. Counter=1
    iv. NewNumber
e) Display FirstNumber, SecondNumber
f) Initiate a loop.
        i. While Counter less than or equal to **Range-2** (Condition for the loop)
        ii. NewNumber= FirstNumber+ SecondNumber
        iii. Display NewNumber
        iv. FirstNumber=SecondNumber.
        SecondNumber=NewNumber
        v. Counter+++; that is counter increases by 1 so far, the condition is fulfilled
        vi. End While Loop
g) Stop

SOLUTION 4:

Bubble sort, sometimes referred to as sinking sort, is a simple sorting algorithm that repeatedly steps through the list or array, compares adjacent elements, and swaps them if they are in the wrong order. The scan through the list is repeated until the list is sorted.

a) Start
b) Enter and read a positive integer for the array size **ArraySize**, ask user to repeat entry for wrong input.
c) Enter and read the elements of the array
d) Declaring integer variables, Let.
    i. **Scan=1; Scan value varies from 1 to ArraySize-1**. For each value of Scan repeat steps **ii to g**
    ii. **SwapPosition=ArraySize-Scan-1; SwapPosition value varies from 1 to ArraySize-Scan-1.** For each value of SwapPosition repeat steps **e to f**
Initiate a loop.
e) Check if **SwapPosition**th element is greater than (**SwapPosition+1)**th. If yes, swap them
f) SwapPosition=SwapPosition+1
g) Scan=Scan+1
h) End the loop
i) Print the sorted array
j) Stop

Note:

$$Maximum\ Number\ of\ scan = ArraySize - 1.$$
$$Position\ of\ the\ swap\ in\ the\ array;\ SwapPosition = ArraySize - Scan - 1$$

SOLUTION 5:

```
a) Start
b) Enter and read a binary number from the user Num, ask user to repeat entry
   for wrong input.
c) Declaring integer variables, Let.
     i.   DecimalNumber=0
     ii.  Power=0
     iii. Reminder
d) If Num=0.
e) Print DecimalNumber
f) Initiate a loop
     i.   Reminder=Num%10…. Modulus of Num, that is the reminder of Num
          when divided by 10
     ii.  Num=Num/10
     iii. DecimalNumber=DecimalNumber+(Reminder×$2^{Power}$)
     vii. Increase Power by 1 so far, the condition is fulfilled
     viii. End the loop
g) Display the DecimalNumber
h) Stop
```