Hands-On Activities – Milestone 2

Activity 1:
Design an algorithm to accept a string from the user and check if it is palindrome or not.

Activity 2:
The sum of the digits of a two-digit number is 15 and the difference between the digits is 3. Design an algorithm to find out the two-digit number.

Activity 3:
Design an algorithm to accept 2 strings **A** and **B** from the user, and check whether any part of string **A** (comprising of minimum of 2 consecutive characters) is a sub-string of string **B**.

For Example -
If string **A** = 1234xyz
and string **B** = abxdekk123

Here the output must be true as 123 is a part of string 'A' which is a sub-string in string 'B'

Note: If no part of string A is present in string B, a message "**No sub strings present**" must be displayed

Activity 4:
The product of two numbers is 9375 and the quotient, when the larger one is divided by the smaller, is 15. Design an algorithm to find the sum of the numbers.

Activity 5:
Design an algorithm to accept the age of 20 students of a class. Re-arrange the data in ascending order (smallest value to largest value). Display the age of the youngest and eldest student in the class. Also display number of students in the class for each age present in the list.

Activity 6:
Design an algorithm to accept a value for 'n' and calculate the total number of all possible squares in a square matrix.

Note: A square matrix is a matrix with the same number of rows and columns, i.e. n x n matrix

For example, if value of n=2, the below 2x2 matrix table contains 5 possible squares –

For example, if value of n=3, the below 3x3 matrix table contains 14 possible squares –

The algorithm must work for any given value of n.

Activity 7:
Design an algorithm to accept 2 strings from the user. Now the algorithm must check if the 2 strings are identical or not. If they are identical the output must be "**IDENTICAL**" else output must be "**NOT IDENTICAL**".

Activity 8:
Design an algorithm to accept 10 numbers from the user. The algorithm must then identify numbers from the set which are divisible by 2, and arrange them in ascending order in a new array. Only numbers which are divisible by 2 must not be present in the new array.

Activity 9:
Design an algorithm to find out how many 3 digit (that is all numbers starting from 100 to 999) numbers are divisible by 6 in all.

Activity 10:
Two goods train each 500 m long, are running in opposite directions on parallel tracks. Their speeds are X km/hr and Y km/hr respectively. Design an algorithm to accept the values of speed X and Y from the user and find the time taken by the slower train to pass the driver of the faster one.
Note: values of X and Y must be positive and non-zero

Activity 11:
The ratio between the length and the breadth of a rectangular park is 3:2. If a man cycling along the boundary of the park at the speed of 12 km/hr completes one round in 8 minutes. Design an algorithm to calculate the area of the park (in sq. m).

Activity 12:

The sum of the squares of three numbers is 138, while the sum of their products taken two at a time is 131. Design an algorithm to find the sum of the 3 numbers.

Activity 13:

A and B together have Rs. 1210. If 4/5 of A's amount is equal to 2/5 of B's amount. Design an algorithm to find out how much amount B has.

Activity 14:

Present ages of Sameer and Anand are in the ratio of 5: 4 respectively. Three years hence, the ratio of their ages will become 11: 9 respectively. Design an algorithm to find out their present ages.

Activity 15:

Design an algorithm for simple calculator which takes input from the user and also details of what operation must be performed. The user can input only 2 operands.
The operations supported by the calculator are +, -, *, /

For Example -
Enter 2 numbers: 3, 5

Enter operation to be performed: +

Output is: 8