

TASKS TO DO:

TASK 1:

You are given the program below: -

```
1  #include <stdio.h>
2
3  void swap(int a, int b){
4      int tmp =a;
5      a = b;
6      b = a;
7  }
8
9  int main(){
10     int j = 2, k = 5;
11     printf("j=%d, k=%d\n", j, k );
12     swap(j,k);
13     printf("j=%d, k=%d\n", j, k );
14     return 0;
15 }
```

- A. Desired result, that is the values of j and k are not swapped. Why is this?
- B. Modify the function swap and its call at line 12 to get the desired result.

Write the answer to question A in the comments of the program for part B.

TASK 2:

Given the function prototype below, implement the function that reverses the array passed to its arguments. Also write the main function that demonstrates this by taking 10 inputs from a user and storing them in an array. Print the array, then use the function reverse and print the array again to show that array has been reversed successfully. Use pointers in the function reverse.

```
void reverse(int *arr, int size){
}
```

Task 3: Write a function for finding factorial of any integer N using recursion.

Task 4: Write a function for finding product of 2 numbers without using multiplication operator and loops. Use recursion and addition only.

Task 5: Euclid gave a recursive description of GCD (HCF) a very long time ago. Here is the simplified version of it.

- 01 If b is 0, then the GCD of a and b is a. This is the base case.
- 02 If b is not 0, then the GCD of a and b is the same as the GCD of b and the remainder when a is divided by b.

Mathematically, $\text{GCD}(a, b) = \text{GCD}(b, a \% b)$, where "%" denotes the remainder when a is divided by b, and $\text{GCD}(a, 0)$ is a.

Write a function for finding GCD first using loops and then using recursion.

Task 6: Find out about Fibonacci series [here](#).

Write recursive function to print n Fibonacci numbers.