

26/11/2025  
Wednesday

(1) Write a C-program to print sum of digits using functions.

```
#include <stdio.h>
int sum of Digits(int n);
int main()
{
    int number, sum;
    printf ("Enter an integer: ");
    if (scanf ("%d", &number) != 1)
    {
        printf ("Invalid input. please enter an integer.\n");
        return 1;
    }
    sum = sum of Digits (number);
    printf ("sum of digits of %d is %d\n", number, sum);
    return 0;
}

int sum of Digits (int n)
{
    int total = 0;
    int remainder;
    if (n<0)
    {
        n = -n;
    }
    while (n!=0)
    {
        remainder = n % 10;
        total += remainder;
        n /= 10;
    }
    return total;
}
```

Output:

Enter an Integer: 12345  
sum of digits of 12345 is 15.

(2) Write a C-program to Reverse numbers using functions.

```
#include <stdio.h> <math.h> stdio.h  
int reverseNumber(int num);  
{  
    int reverse = 0, remainder; // name for?  
    while (num != 0)  
    {  
        remainder = num % 10; // 01 = 1 mod 10  
        reverse = reverse * 10 + remainder; // 02 = 0 mod 10  
        num /= 10; // 03 = 0 mod 10  
    }  
    return reverse;  
}  
int main()  
{  
    int num, reversed;  
    printf("Enter a number: ");  
    scanf("%d", &num);  
    reversed = reverseNumber(num);  
    printf("Original number : %d\n", num);  
    printf("Reversed number: %d\n", reversed);  
    return 0;  
}
```

Output:

Enter a number : 12345

Original number: 12345

Reversed Number: 54321

(3) Swap two numbers without swapping using third variable.

```
#include <stdio.h>
void swapNumbers(int *a, int *b);
int main()
{
    int num1 = 10;
    int num2 = 20;
    printf("... Before swapping ... \n");
    printf("First number (num1): %d \n", num1);
    printf("Second number (num2): %d \n", num2);

    Swap Numbers (&num1, &num2);
    printf("... After swapping ... \n");
    printf("First number (num1): %d \n", num1);
    printf("Second number (num2): %d \n", num2);
    return 0;
}

void swapNumbers(int *a, int *b)
{
    (*a = *a + *b);
    (*b = *a - *b);
    (*a = *a - *b);
}
```

Output:-

--- Before Swapping ---

First Number (num1): 10

Second Number (num2): 20

--- After Swapping ---

First Number (num1): 20

Second Number (num2): 10