Function Assignment

Write the below programs by writing a separate function and calling it in the main function and implement **proper ERROR Handling** wherever necessary

1) Write a function to calculate simple interest. Call it in main function with appropriate inputs and print the total amount the user will get after the tenure (principle + interest)

Solution: #include <stdio.h>

```
int main() {
float principal, rate, time, interest, total_amount;
// Input values
printf("Enter principal amount: ");
scanf("%f", &principal);
printf("Enter rate of interest (in %%): ");
scanf("%f", &rate);
printf("Enter time (in years): ");
scanf("%f", &time);
// Calculate simple interest
interest = (principal * rate * time) / 100;

// Calculate total amount
total_amount = principal + interest;

// Print results
printf("Simple Interest: %.2f\n", interest);
printf("Total Amount after %.2f years: %.2f\n", time, total_amount);
return 0;
}
```

Output: Enter principal amount: 1000

Enter rate of interest (in %): 5

Enter time (in years): 2

Simple Interest: 100.00

Total Amount after 2.00 years: 1100.00

2. Write a function that takes two numbers, a and n as input arguments and returns the value of a to the power of n.

Solution: #include <stdio.h>

```
double power(double a, int n) {
    double result = 1;
    for (int i = 0; i < n; i++) {
        result *= a;
    }
    return result;
}
int main() {
    double a;
    int n;
    printf("Enter base (a): ");
    scanf("%lf", &a);
    printf("Enter exponent (n): ");
    scanf("%d", &n);
    printf("%.2lf^%d = %.2lf\n", a, n, power(a, n));
    return 0;
}
```

3) Write a function that takes two numbers a and b as input arguments and returns their product as return value , without using * operator.

```
#include <stdio.h>
int multiply(int a, int b) {
   int result = 0;
   for (int i = 0; i < b; i++) {
      result += a;  //
   }
   return result;
}
int main() {
   int a, b;
   printf("Enter two numbers: ");
   scanf("%d %d", &a, &b);
   int product = multiply(a, b);
   printf("Product: %d\n", product);
   return 0;
}
```

Input: Enter two numbers: 4 3

Output: Product: 12

4) Write a function that takes two numbers a and b, and returns the quotient after dividing a with b

```
#include <stdio.h>
int divide(int a, int b) {
   if (b == 0) {
      printf("Error! Division by zero is not allowed.\n");
```

```
        return 0; //
    }
    return a / b; //
}

int main() {
    int a, b;
    printf("Enter two numbers (a and b): ");
    scanf("%d %d", &a, &b);

    int quotient = divide(a, b);
    printf("Quotient: %d\n", quotient);

    return 0;
}
```

Input: Enter two numbers (a and b): 10 , 2

Output: 5

5) Write a function that takes two numbers a and b, and returns the remainder after dividing a with b

```
#include <stdio.h>
int getRemainder(int a, int b) {
    if (b == 0) {
        printf("Error! Division by zero is not allowed.\n");
        return 0; //
    }
    return a % b; //
```

```
}

int main() {
    int a, b;
    printf("Enter two numbers (a and b): ");
    scanf("%d %d", &a, &b);

    int remainder = getRemainder(a, b);
    printf("Remainder: %d\n", remainder);

    return 0;
}
```

Input: Enter two numbers (a and b): 10 3

Output: Remainder: 1

6) Write a function that takes an integer number as input and prints its multiplicationtable.return type is void

```
#include <stdio.h>
void print Multiplication Table(int num) {
    printf("Multiplication Table of %d:\n", num);
    for (int i = 1; i <= 10; i++) {
        printf("%d x %d = %d\n", num, i, num * i);
    }
}
int main() {
    int num;
    printf("Enter a number: ");
```

```
    scanf("%d", &num);
    print Multiplication Table(num);
    return 0;
}
```

Input: Enter a number: 5

Output: Multiplication Table of 5:

5 x 1 = 5

5 x 2 = 10

5 x 3 = 15

5 x 4 = 20

5 x 5 = 25

5 x 6 = 30

5 x 7 = 35

5 x 8 = 40

5 x 9 = 45

5 x 10 = 50

7) write a function that can take an integer as input and return 1 if the number is prime number, return 0 if it is not prime and print appropriate output message in main according to output. return type is integer. function name Is Prime - returns int (0 or 1)

```
#include <stdio.h>
int Is Prime(int num) {
    if (num < 2) return 0;
    for (int i = 2; i * i <= num; i++)
        if (num % i == 0) return 0;
    return 1;
}
```

```
int main() {
    int num;
    printf("Enter a number: ");
    scanf("%d", &num);
    printf(num % 2 == 0 ? "%d is not a prime number.\n" : "%d is a prime number.\n", num, Is
Prime(num) ? num : num);
    return 0;
}
```

Input: Enter a number: 7

Output: 7 is a prime number.

8).Write a function to determine if a character is alphanumeric or not and print the appropriate output in main function. (return 1 if it is alphanumeric, 0 if it is not alpha numeric).

```
#include <stdio.h>
int fun_alpha_num(char c) {
    return ((c >= 'A' && c <= 'Z') || (c >= 'a' && c <= 'z') || (c >= '0' && c <= '9'));
}
int main() {
    char c;
    printf("Enter a character: ");
    scanf(" %c", &c);
    if (fun_alpha_num(c))
        printf("%c is alphanumeric.\n", c);
   else
        printf("%c is not alphanumeric.\n", c);
    return 0;
```

}

Input: Enter a character: A

Output: A is alphanumeric.

9) Write a calculator program with 5 functions below to do the operations of addition, subtraction, multiplication, division for quotient and division for remainder.

1) int add (int a, int b) // to be called when user chooses +

2) int sub (int a,int b) // to be called when user chooses -

3) int mul(int a, int b) // to be called when user chooses *

4) int quotient(int a, int b) // to be called when user chooses /

5) int remainder(int a,int b) // to be called when user choose%

```
#include <stdio.h>
int add(int a, int b) { return a + b; }
int sub(int a, int b) { return a - b; }
int mul(int a, int b) { return a * b; }
int quotient(int a, int b) { return b ? a / b : 0; }
int remainder(int a, int b) { return b ? a % b : 0; }
int main() {
   int a, b;
   char op;
   printf("Enter two numbers and operator (+, -, *, /, %%): ");
   scanf("%d %d %c", &a, &b, &op);
   switch(op) {
      case '+': printf("%d\n", add(a, b)); break;
      case '-': printf("%d\n", sub(a, b)); break;
      case '*': printf("%d\n", mul(a, b)); break;
      case '/': printf("%d\n", b ? quotient(a, b) : 0); break;
```

```
        case '%': printf("%d\n", b ? remainder(a, b) : 0); break;
        default: printf("Invalid operator\n");
    }
    return 0;
}
```

Input: Enter two numbers and operator: 10*2

Output:20

10) Write a function to accept a year as input and return 1 if the year is a leap year, otherwise 0.

```
#include <stdio.h>
int isLeapYear(int year) {
    return (year % 4 == 0 && year % 100 != 0) || (year % 400 == 0);
}
int main() {
    int year;
    printf("Enter a year: ");
    scanf("%d", &year);
    if (isLeapYear(year))
        printf("%d is a leap year.\n", year);
    else
        printf("%d is not a leap year.\n", year);
    return 0;
}
```

Input: Enter a year: 2024

Output:2024 is a leap year

11).Write a function to accept a month and year as input, and return the number of days in that month as output. print the number of days in main.

```
#include <stdio.h>
int daysInMonth(int month, int year) {
   if (month == 2)
      return (year % 4 == 0 && year % 100 != 0) || (year % 400 == 0) ? 29 : 28;
   return (month == 4 || month == 6 || month == 9 || month == 11) ? 30 : 31;
}
int main() {
   int month, year;
   printf("Enter month and year: ");
   scanf("%d %d", &month, &year);
   printf("Days: %d\n", daysInMonth(month, year));
   return 0;
}
```

Input: Enter month and year: 2 2024

Output: Days: 29

12).Write a function ,that can take two integers, swap their values and print their new values. return type should be void. function should print the values.

```
#include <stdio.h>
void swap(int a, int b) {
   int temp = a;
   a = b;
   b = temp;
   printf("After swap: a = %d, b = %d\n", a, b);
}
```

```
int main() {
   int a, b;
   printf("Enter two numbers: ");
   scanf("%d %d", &a, &b);
   swap(a, b);
   return 0;
}
```

Input: Enter two numbers: 5 10

Output: After swap: a = 10, b = 5

13).Write a function that takes 2 dates as inputs, and returns 1 if the first date is old, 2 if the second date is old

```
#include <stdio.h>
int compareDates(int d1, int m1, int y1, int d2, int m2, int y2) {
   if (y1 < y2 || (y1 == y2 && m1 < m2) || (y1 == y2 && m1 == m2 && d1 < d2))
      return 1;
   return 2;
}
int main() {
   int d1, m1, y1, d2, m2, y2;
   printf("Enter first date (DD MM YYYY): ");
   scanf("%d %d %d", &d1, &m1, &y1);
   printf("Enter second date (DD MM YYYY): ");
   scanf("%d %d %d", &d2, &m2, &y2);

  int result = compareDates(d1, m1, y1, d2, m2, y2);
   printf("Date %d is older.\n", result);
```

```
  return 0;
}
```

Input: Enter first date (DD MM YYYY): 10 5 2020

Enter second date (DD MM YYYY): 15 6 2023

Output: Date 1 is older.

14).Write a function that takes a date as input, and returns 1 if the date is valid, 0 if the date is valid

```
#include <stdio.h>
int isValidDate(int d, int m, int y) {
   if (m < 1 || m > 12 || d < 1 || y < 1) return 0;
   int daysInMonth[] = {31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};
   if ((y % 4 == 0 && y % 100 != 0) || (y % 400 == 0)) daysInMonth[1] = 29;
   return d <= daysInMonth[m - 1];
}
int main() {
   int d, m, y;
   printf("Enter a date (DD MM YYYY): ");
   scanf("%d %d %d", &d, &m, &y);
   if (isValidDate(d, m, y))
      printf("Valid date.\n");
   else
      printf("Invalid date.\n");

 return 0;
}
```

Input: Enter a date (DD MM YYYY): 29 02 2024

Output: valid date