CSE ASSIGNMENT 2

Problem 1

Question:

> *Write a C program to find factorial of 'n' numbers*

Code:

```c
// Write a C program to calculate the factorial
// To instantly test this code, copy it and go over to https://replit.com/languages/c


// Include the input and output helper header file
#include <stdio.h>

// Code starts execution from header
int main()
{
    // Store the number and the final value storage
    int num, fact;

    // Get input and store it in the var
    // {
    printf("Enter the number to find factorial : ");
    scanf("%d", &num);
    // }

    // Find the factorial and print it out
    // {
    fact = 1;

    if (num == 0 || num == 1)
    {
    }
    else
    {
        while (num != 1)
        {
            fact *= num;
            num--;
        }
    }

    printf("The factorial was : %d", fact);
    // }

    // Return 0 to indicate code ran with no errors
```

```
    return 0;
}
```

Output:

```
Enter the number to find factorial : 5
The factorial was : 120
```

Problem 2

Question:

> *Write a C program to generate fibonacci series of 'n' numbers*

Code:

```
// Write a C program to print out `n` fibonacci terms
// To instantly test this code, copy it and go over to https://replit.com/languages/c


// Include the input and output helper header file
#include <stdio.h>

// Code starts execution from header
int main()
{
    // Number of terms, t1, t2, and current value
    int num, t1, t2, curr;

    // Get the number of terms required
    // {
    printf("Enter the number of terms required (num > 3) : ");
    scanf("%d", &num);
    // }

    // Print first two terms
    // {
    t1 = 0;
    t2 = 1;

    printf("Fibonacci Series :\n%d %d", t1, t2);
    // }

    // Print out the remaining terms
    while (num - 2 > 0)
    {
        curr = t1 + t2;

        printf(" %d", curr);
```

```
        t1 = t2;
        t2 = curr;
        num--;
    }

    // Return 0 to show success
    return 0;
}
```

Output:

```
Enter the number of terms required (num > 3) : 5
Fibonacci Series :
0 1 1 2 3
```

Problem 3

Question:

> *Write a C program to calculate sum of different 'n' numbers*

Code:

```
// Write a C program to calculate sum of `n` number got from the user
// To instantly test this code, copy it and go over to https://replit.com/languages/c


// Include the input and output helper header file
#include <stdio.h>

// Code starts execution from header
int main()
{
    // The number of numbers and sum and temp
    int n, sum, temp;

    // Get the number of numbers the user wants to add
    // {
    printf("Enter the number of numbers you would like to add : ");
    scanf("%d", &n);
    // }

    // Get the inputs and print out the sum
    // {
    sum = 0;

    while (n != 0)
    {
        printf("Enter a number : ");
        scanf("%d", &temp);
```

```
        sum += temp;
        n--;
    }

    printf("The sum of all the numbers is : %d", sum);
    // }

    // Return 0 to show successful run
    return 0;
}
```

Output:

```
Enter the number of numbers you would like to add : 4
Enter a number : 1
Enter a number : 1
Enter a number : 2
Enter a number : 2
The sum of all the numbers is : 6
```

Problem 4

Question:

> *Write a C program to Print the divisors of the given positive integer 'n'*

Code:

```
// Write a C program to print out the divisors of a number `n`
// To instantly test this code, copy it and go over to https://replit.com/languages/c


// Include the input and output helper header file
#include <stdio.h>

// Code starts execution from header
int main()
{
    // Number
    int n;

    // Get the input
    // {
    printf("Enter a number : ");
    scanf("%d", &n);
    // }

    // Print the factors
    // {
```

```
    printf("Factors :\n1");

    for (int i = 2; i != n; i++)
    {
        if (n % i == 0)
        {
            printf(" %d", i);
        }
    }

    printf(" %d", n);
    // }

    // Return 0 to show success
    return 0;
}
```

Output:

```
Enter a number : 12
Factors :
1 2 3 4 6 12
```

Problem 5

Question:

> *Write a C program to find GCD and LCM of two numbers*

Code:

```
// Write a C program to find GCD and LCM of two numbers
// To instantly test this code, copy it and go over to https://replit.com/languages/c


// Include the input and output helper header file
#include <stdio.h>

// Code starts execution from header
int main()
{
    // Storage vars
    int num1, num2, gcd, lcm, remainder, numerator, denominator;

    // Getting input
    // {
    printf("Enter two numbers : ");
    scanf("%d %d", &num1, &num2);
    // }
```

```c
    // Compute GCD and LCM and print it
    // {
    if (num1 > num2)
    {
        numerator = num1;
        denominator = num2;
    }
    else
    {
        numerator = num2;
        denominator = num1;
    }

    remainder = numerator % denominator;

    while (remainder != 0)
    {
        numerator   = denominator;
        denominator = remainder;
        remainder   = numerator % denominator;
    }

    gcd = denominator;
    lcm = num1 * num2 / gcd;

    printf("GCD of %d and %d = %d\n", num1, num2, gcd);
    printf("LCM of %d and %d = %d\n", num1, num2, lcm);
    // }

    // Return 0 to show success
    return 0;
}
```

Output:

```
Enter two numbers : 5 17
GCD of 5 and 17 = 1
LCM of 5 and 17 = 85
```

Problem 6

Question:

> *Write a C program to convert decimal to binary, octal and hexadecimal numbers using switch case statement*

Code:

```c
// Write a C program to convert decimal to binary, octal and hexadecimal
// To instantly test this code, copy it and go over to https://replit.com/languages/c
```

```c
// Include the input and output helper header file
#include <stdio.h>

// Code starts execution from header
int main()
{
    // User input, choice and array
    int num, choice, arr[11] = {0, 0 ,0 ,0 ,0 ,0 ,0 ,0 ,0, 0};

    // Get the value and choice
    // {
    printf("Enter a value : ");
    scanf("%d", &num);
    printf("Enter a choice (1, 2, 3) : ");
    scanf("%d", &choice);
    // }

    // Compute equivalent based on choice
    switch (choice)
    {
        case 1: // Convert to binary

            for (int i = 31; i >= 0; i--)
            {
                if ((num >> i) & 1)
                {
                    printf("1");
                }
                else
                {
                    printf("0");
                }
            }

            break;

        case 2: // Convert to Octal

            for (int i = 0; num != 0; i++)
            {
                arr[i] = num % 8;
                num /= 8;
            }

            for (int j = 10; j >= 0; j--)
            {
                printf("%d", arr[j]);
            }

            break;
```

```
        case 3: // Convert to Hex

            for (int i = 0; num != 0; i++)
            {
                arr[i] = num % 16;
                num /= 16;
            }

            for (int j = 7; j >= 0; j--)
            {
                char map[] = "ABCDEF";

                if (arr[j] / 10 > 0)
                {
                    printf("%c", map[(arr[j] % 9) - 1]);
                }
                else
                {
                    printf("%d", arr[j]);
                }
            }

            break;
    }

    // Return 0 to show success
    return 0;
}
```

Output 1:

```
Enter a value : 5000
Enter a choice (1, 2, 3) : 1
00000000000000000001001110001000
```

Output 2:

```
Enter a value : 5000
Enter a choice (1, 2, 3) : 2
00000011610
```

Output 3:

```
Enter a value : 5000
Enter a choice (1, 2, 3) : 3
00001388
```
