Assignment - 24 **A Job Ready Bootcamp in C++, DSA and IOT** **MySirG**

Functions in C++

1. Define a function to check whether a given number is a Prime number or not.

```
#include <iostream>
using namespace std;
void checkPrime(int);
int main()
{
    int n;
    cout << "Enter a number to check Prime number or not : ";
    cin >> n;
    checkPrime(n);
    return 0;
}
void checkPrime(int n)
{
    int i = 2;
    while (i <= n / 2)
    {
        if (n % i == 0)
        {
            break;
        }
        i++;
    }
    i == ((n / 2) + 1) ? cout << n << " is a prime number" : cout << n << "
is not a prime number";
}
==============================================================================
Output:
Enter a number to check Prime number or not : 19
19 is a prime number
```

2. Define a function to find the highest value digit in a given number.

```
#include <iostream>
using namespace std;
int highestDigit(int);
int main()
{
    int n;
    cout << "Enter a number : ";
    cin >> n;
    cout << "The highest value digit in a given number is " <<
highestDigit(n);
    return 0;
}
int highestDigit(int n)
{
    int rem, temp = 0;
    while (n)
    {
        rem = n % 10;
        n = n / 10;
        if (temp < rem)
            temp = rem;
    }
    return temp;
}
==============================================================================
Output:
Enter a number : 42351
```

```
The highest value digit in a given number is 5
```

3. Define a function to calculate x raised to the power y.

```
#include <iostream>
using namespace std;
int powerCalculatio(int, int);
int main()
{
    int number, power;
    cout << "Enter a number : ";
    cin >> number;
    cout << "Enter " << number << " to the power ";
    cin >> power;
    cout << number << " to the power " << power << " is " <<
powerCalculatio(number, power);
    return 0;
}

int powerCalculatio(int x, int y)
{
    int i, result = x;
    if (y == 0)
        return 1;
    else if (y == 1)
        return x;
    else
        for (i = 1; i < y; i++)
            result = result * x;
    return result;
}
==============================================================================
Output:
Enter a number : 5
Enter 5 to the power 4
5 to the power 4 is 625
```

4. Define a function to print Pascal Triangle up to N lines.

```
#include <iostream>
using namespace std;
int fact(int);
void pascalTriangle(int);
int main()
{
    int n;
    cout<<"Enter number of rows to print PASCAL Triangle : ";
    cin>>n;
    pascalTriangle(n);
    return 0;
}
// below function is to print pascal triangle
void pascalTriangle(int x)
{
    for (int i = 0; i < x; i++)
    {
        for (int j = 0; j <= i; j++)
        {
            if (j <= i)
                cout<<fact(i) / (fact(j) * fact(i - j))<<"  ";
            else
                cout<<" ";
        } // end inner loop
        cout<<endl;
    } // end outer loop
}
// below function is to calculate factorial
int fact(int a)
{
    int i = 1, s = 1;
    while (i <= a)
    {
        s = s * i;
        i++;
    }
    return (s);
}
=============================================================================
Output:
Enter number of rows to print PASCAL Triangle : 5
1
1  1
1  2  1
1  3  3  1
1  4  6  4  1
```

5. Define a function to check whether a given number is a term in a Fibonacci series or not.

```
#include <iostream>
using namespace std;
void checkTerm(int);
int main()
{
    int x;
    cout << "Enter a number : ";
    cin >> x;
    checkTerm(x);
    return 0;
}
void checkTerm(int n)
{
    int i, a = -1, b = 1, s = 0;
    for (i = 0; i <= n + 2; i++)
    {
        s = a + b;
        if (n == s)
        {
            cout << "Given number is a term in a Fibonacci series";
            break;
        }
        if (s > n)
        {
            cout << "Given number is not in a Fibonacci series";
            break;
        }
        a = b;
        b = s;
    }
}
=============================================================================
Output:
Enter a number : 4181
Given number is a term in a Fibonacci series
```

6. Define a function to swap data of two int variables using call by reference

```
#include <iostream>
using namespace std;
void swap(int &, int &);
int main()
```

```
{
    int a, b;
    cout << "Enter two numbers : ";
    cin >> a >> b;
    cout << "Before Swapping" << endl;
    cout << "a    :   " << a << endl;
    cout << "b    :   " << b << endl;
    swap(a, b);
    cout << "After Swapping" << endl;
    cout << "a    :   " << a << endl;
    cout << "b    :   " << b << endl;
    return 0;
}
void swap(int &x, int &y)
{
    int temp;
    temp = x;
    x = y;
    y = temp;
}
=============================================================================
Output:
Enter two numbers : 50 20
Before Swapping
a    :   50
b    :   20
After Swapping
a    :   20
b    :   50
```

7. Write a function using the default argument that is able to add 2 or 3 numbers.

```
#include <iostream>
using namespace std;
int add(int a, int = 10, int = 5);
int main()
{
    int x, y, z;
    cout << "Enter two numbers : ";
    cin >> x >> y;
    cout << "Sum is "<< add(x, y) << endl;
    cout << "Enter three numbers : ";
    cin >> x >> y >> z;
    cout << "Sum is " << add(x, y, z);
    return 0;
}
int add(int a, int b, int c)
{
    return (a + b + c);
}
=============================================================================
Output:
Enter two numbers : 10 20
Sum is 35
Enter three numbers : 10 20 30
Sum is 60
```

8. Define overloaded functions to calculate area of circle, area of rectangle and area of triangle

```
#include <iostream>
using namespace std;
void area(float);
void area(float, float);
void area();
```

```
int main()
{
    float r, l, b;
    cout << "Enter radius of a circle : ";
    cin >> r;
    area(r);    // area of circle
    cout << "Enter length and breadth of a rectangle : ";
    cin >> l >> b;
    area(l, b); // area of rectangle
    area(); // area of triangle
    return 0;
}
//below function, area of circle
void area(float r)
{
    cout << "Area of a circle is " << (3.14159 * r * r) << " unit" << endl;
}
//below function, area of rectangle
void area(float l, float b)
{
    cout << "Area of rectangle is " << l * b << " unit" << endl;
}
//below function, area of triangle
void area()
{
    float b, h;
    cout << "Enter base and height of a triangle : ";
    cin >> b >> h;
    cout << "Area of a triangle is " << (b * h) / 2 << " unit" << endl;
}
==============================================================================
Output:
Enter radius of a circle : 8.5
Area of a circle is 226.98 unit
Enter length and breadth of a rectangle : 12.21 14.63
Area of rectangle is 178.632 unit
Enter base and height of a triangle : 55.24 64.25
Area of a triangle is 1774.59 unit
```

9. Write functions using function overloading to find a maximum of two numbers and both the numbers can be integer or real.

```
#include <iostream>
using namespace std;

void max(int , int);
void max(double , double);
void max(double, int);
void max(int, double);

int main()
{

    max(8, 2);
    max(9.23, 7.52);
    max(9.23, 12);
    max(15, 7.52);

    return 0;
}

void max(int a, int b)
{
    a > b ? cout << a << " is greater" << endl : cout << b << " is greater"
<< endl;
}

void max(double x, double y)
{
    x > y ? cout << x << " is greater" << endl : cout << y << " is greater"
<< endl;
}
void max(int a, double x)
{
    a > x ? cout << a << " is greater" << endl : cout << x << " is greater"
<< endl;
}
void max(double a, int x)
{
    a > x ? cout << a << " is greater" << endl : cout << x << " is greater"
<< endl;
}
=============================================================================
Output:
8 is greater
9.23 is greater
12 is greater
15 is greater
```

10. Write functions using function overloading to add two numbers having different data types.

```
#include <iostream>
using namespace std;

void add(int, int);
void add(double, double);
void add(double, int);
void add(int, double);

int main()
{

    add(8, 2);
    add(9.23, 7.52);
    add(9.23, 12);
    add(15, 7.52);

    return 0;
}

void add(int a, int b)
{
    cout << "Sum of " <<a<<" and "<< b << " is "<< a + b << endl;
}

void add(double x, double y)
{
    cout << "Sum of " <<x<<" and "<< y << " is "<< x + y << endl;
}
void add(int a, double x)
{
    cout << "Sum of " <<a<<" and "<< x << " is "<< a + x << endl;
}
void add(double a, int x)
{
    cout << "Sum of " <<a<<" and "<< x << " is "<< a + x << endl;
}
=============================================================================
Output:
Sum of 8 and 2 is 10
Sum of 9.23 and 7.52 is 16.75
Sum of 9.23 and 12 is 21.23
Sum of 15 and 7.52 is 22.52
```