1. Convert the following while loop to a do-while loop:

```
int x = 1;
while (x > 0)
{
    cout << "enter a number: ";
    cin >> x;
}
```

Code:

```
#include <iostream>
using namespace std;
int main()
{
    int x;
    do
    {
          cout << "enter a number: ";
          cin >> x;
    }while (x > 0);
    cout << endl;
    return 0;
}
```

Output:

```
enter a number: 34
enter a number: 56
enter a number: -3
```

2. Write a do-while loop that asks the user to enter two numbers and operator +, -, / ,*. After performing operation, and the resultant answer should be displayed, and the user should be asked if he or she wishes to perform the operation again. If so, the loop should repeat; otherwise it should terminate.

Code:

```
#include <iostream>

using namespace std;
int main()
{
    float num1, num2, result = 0;
    char op, repeat;
    do
    {
          cout << "\nEnter the first number: ";
          cin >> num1;
          cout << "Enter the second number: ";
          cin >> num2;
          cout << "Enter the operator: ";
          cin >> op;
          switch (op)
          {
          case '+':
                result = num1 + num2;
                cout << "Sum of two numbers: " << result;
                break;
          case '-':
                result = num1 - num2;
```

```
                cout << "Difference of two numbers: " << result;
                break;
        case '*':
                result = num1 * num2;
                cout << "Product of two numbers: " << result;
                break;
        case '/':
                result = num1 / num2;
                cout << "Division of two numbers: " << result;
                break;
        default:
                cout << "Invalid Input!";
        }
        cout << "\nDo you want to perform operation again press Y/y
otherwise press N/n ";
        cin >> repeat;
    }while (repeat == 'Y' || repeat == 'y');
    cout << endl;
    return 0;
}
```

Output:

```
Enter the first number: 23
Enter the second number: 12
Enter the operator: -
Difference of two numbers: 11
Do you want to perform operation again press Y/y otherwise press N/n y

Enter the first number: 45
Enter the second number: 5
Enter the operator: /
Division of two numbers: 9
Do you want to perform operation again press Y/y otherwise press N/n y

Enter the first number: 46
Enter the second number: 7
Enter the operator: /
Division of two numbers: 6.57143
Do you want to perform operation again press Y/y otherwise press N/n n
```

3. Write a C++ program to Print Table of any Number using while loop.

   Code:

```
#include <iostream>
using namespace std;
int main()
{
    int num, i = 1;
    cout << "Enter any number: ";
    cin >> num;
    while (i <= 10)
    {
        cout << num << " * " << i << " = " << num * i << endl;
        i++;
    }
    cout << endl;
    return 0;
}
```

   Output:

```
Enter any number: 19
19 * 1 = 19
19 * 2 = 38
19 * 3 = 57
19 * 4 = 76
19 * 5 = 95
19 * 6 = 114
19 * 7 = 133
19 * 8 = 152
19 * 9 = 171
19 * 10 = 190
```

4. Write a C++ program, take two integer i (starting point) and n (ending point) from user and print all even numbers between i to n using while loop. For example: Let say i=5; and n =20; Output should be (6 8 10 12 14 16 18 20)
   Code:

```
#include <iostream>
using namespace std;
int main()
{
    int n, i;
    cout << "Enter the starting integer: ";
    cin >> i;
    cout << "Enter the ending integer: ";
    cin >> n;
    if (i % 2 == 0)
    {
          cout << "Even number between " << i << " and " << n << " is:
" << i << " ";
    }
    else
    {
          cout << "Even number between " << i << " and " << n << " is:
";
    }

    while (i <= n-2)
    {
          if (i % 2 == 0)
          {
                i += 2;
          }
          else
          {
                i++;
          }
          cout << i << " ";
    }
    cout << endl;
    return 0;
}
```

   Output:

```
Enter the starting integer: 2
Enter the ending integer: 9
Even number between 2 and 9 is: 2 4 6 8
```

5. Write a C++ program to print natural numbers in reverse from n to 1 using both do while loop and while loop.

   Code:

   Using the while loop:

```cpp
#include <iostream>
using namespace std;
int main()
{
    int n, i;
    cout << "Enter the any natural number: ";
    cin >> n;

    cout << "The reverse series is: ";

    while (n >= 1)
    {
          cout << n << " ";
          n--;
    }
    cout << endl;
    return 0;
}
```

   Output:

```
Enter the any natural number: 13
The reverse series is: 13 12 11 10 9 8 7 6 5 4 3 2 1
```

   Code:

   Using the do while loop:

```cpp
#include <iostream>
using namespace std;
int main()
{
    int n, i;
    cout << "Enter the any natural number: ";
    cin >> n;

    cout << "The reverse series is: ";
    do
    {
          n--;
          cout << n << " ";
    } while (n >= 1);
    cout << endl;
    return 0;
}
```

   Output:

```
Enter the any natural number: 13
The reverse series is: 13 12 11 10 9 8 7 6 5 4 3 2 1
```

6. Write a program in C++ to find LCM of any two numbers using HCF.

Code:

```
#include <iostream>
using namespace std;
int main()
{
	int num1, num2, r1, r2, hcf, lcm;
	cout << "Enter first number: ";
	cin >> num1;
	cout << "Enter second number: ";
	cin >> num2;
	for (int i = 1; i <= num1; i++)
	{
		r1 = num1 % i;
		r2 = num2 % i;
		if (r1 == 0 && r2 == 0)
		{
			hcf = i;
		}
	}
	lcm = (num1 * num2) / hcf;
	cout <<"The LCM of " <<num1 <<" and "<<num2 << " is: " << lcm;
	cout << endl;
	return 0;
}
```

Output:

```
Enter first number: 32
Enter second number: 48
The LCM of 32 and 48 is: 96
```

1. Write a program in C++ to find out the sum of an Arithmetic progression series.

Code:

```
#include <iostream>
using namespace std;
int main()
{
	int n, a, d, sum = 0;
	cout << "Enter the number of terms in the series: ";
	cin >> n;
	cout << "Enter the first term of the series: ";
	cin >> a;
	cout << "Enter the common difference: ";
	cin >> d;
	for (int i = 0; i < n; i++)
	{
		sum += a + i * d;
	}
	cout << "Sum of arithmetic progression series: " << sum;
	cout << endl;
	return 0;
}
```

Output:

```
Enter the number of terms in the series: 10
Enter the first term of the series: 1
Enter the common difference: 2
Sum of arithmetic progression series: 100
```

2. Write a program in C++ to create a diamond.
   Code:

```cpp
#include <iostream>
using namespace std;
int main()
{
	for (int i = 0; i <= 8; i++)
	{
		for (int j = 0; j <= 8; j++)
		{
			if (i <= 4)
			{
				if (j >= 4 - i && j <= 4 + i)
				{
					cout << "*";
				}
				else
				{
					cout << " ";
				}
			}
			else
			{
				if (j >= i - 4 && j <= 12 - i)
				{
					cout << "*";
				}
				else
				{
					cout << " ";
				}
			}
		}
		cout << endl;
	}
	cout << endl;
	return 0;
}
```

   Output:

```
    *
   ***
  *****
 *******
*********
 *******
  *****
   ***
    *
```

3. Write a program in C++ to convert a decimal number to binary number.

   Code:

```cpp
#include <iostream>
using namespace std;
int main()
{
	int num, q, r;
	cout << "Enter the decimal number: ";
	cin >> num;
	do
	{
		q = num / 2;
		r = num % 2;
		num = q;
		cout << r;
	} while (q > 1);
	cout << q;
	cout << endl;
	return 0;
}
```

   Output:

4. Write a program in C++ to convert a binary number to decimal number.

   Code:

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

int main()
{
    int d_num, b_num = 0, place = 1;

    cout << "Enter a decimal number: ";
    cin >> d_num;

    if (d_num == 0)
    {
        cout << "Binary representation: 0" << endl;
        return 0;
    }

    while (d_num > 0)
    {
        int bit = d_num % 2;
        b_num += bit * place;
        place *= 10;
        d_num /= 2;
    }
    cout << "Binary representation: " << b_num << endl;

    return 0;
}
```

   Output:

```
Enter a decimal number: 23
Binary representation: 10111
```

5. Write a program in C++ to convert a binary number to decimal number.
   Code:

```cpp
#include <iostream>

using namespace std;

int main() {
    long long b_num;
    int d_num = 0, power = 1;
    cout << "Enter a binary number: ";
    cin >> b_num;

    while (b_num != 0)
    {
        int digit = b_num % 10;
        d_num += digit * power; // Corrected this line
        power *= 2;
        b_num /= 10;
    }

    cout << "Decimal representation: " << d_num << endl;

    return 0;
}
```

   Output: