1. Write a function minmax(int a, int b) that will take two integers and simultaneously return both their min and their max. Use structures to return the values.
2. Write a function arraysum(int n, int a[], int b[]) that will take two arrays of equal length n, and find their indexwise sum. For example, if a = {2, 3, 4} and b = {1, 2, 3}, their index-wise sum would be {3, 5, 7}. Use structures to return the values.
3. Remember the Fraction structure that we wrote. Using it write functions to
   1. Subtract two fractions
   2. Multiply Two fractions
   3. Divide two fractions

1. Remember that we wrote a program to convert Fahrenheit into Celcius. Now do the same but using fractions. Your input and output both should be in fractions.

1. Write a function that compares two functions and checks if they have equal values. For example - are all equal.
2. Write a C program to find the lowest set bit of a given number using bitwise operators. For example 28 has binary representation 11100. And the lowest set bit at index 2.

**Input**

Input any number: 28

**Output**

Lowest set bit in 28 is at position 2.

1. Write a C program count total number of ones (1s) and zeros (0s) in given **int** variable.

**Example**

**Input**

Input any number: 22

**Output**

Output number of ones: 3

Output number of zeros: 29

1. Remember the RightRotate function we wrote that right rotates the bits of a number. Write a LeftRotate function that rotates bits to the left. Your function should have the signature unsigned char LeftRotate(unsigned char x, int k);

1. Now combine the above LeftRotate and RightRotate in a function

unsigned char Rotate(unsigned char x, int k);

When k is positive, the function should rotate Right By k bits. When k is negative, it should rotate left by abs(k) bits.

1. How does the LeftRotate and RightRotate functions change when the input type changes from unsigned char to unsigned int.