****BITWISE operator Exercises ::::::::::::::::::::::::::::::::::::;;

1.

```
#include <stdio.h>
int main()
{
   int n;
   printf("Input a number: ");
   scanf("%d", &n);
   if(n & 1)
      printf("LSB of %d is set (1).", n);
   else
      printf("LSB of %d is unset (0).", n);

   return 0;
}
```

2.

```
#include <stdio.h>
int main()
{
   int num, msb,BITS;
   printf("Enter any number: ");
   scanf("%d", &num);
   BITS=sizeof(num)*8;
   msb = 1 << (BITS - 1);
   if(num & msb)
      printf("MSB of %d is set (1).", num);
   else
      printf("MSB of %d is unset (0).", num);

   return 0;
}
```

3.

```
#include <stdio.h>
int main()
{
   int num, n, bitStatus;
   printf("Input any number: ");
   scanf("%d", &num);
   printf("Enter nth bit to check (0-31): ");
   scanf("%d", &n);
   bitStatus = (num >> n) & 1;
   printf("The %dth bit is set to %d", n, bitStatus);
   return 0;
}
```

4.

```
#include <stdio.h>
int main()
{
   int num, n, newNum;
   printf("Enter any number: ");
   scanf("%d", &num);
   printf("Enter nth bit to set (0-31): ");
   scanf("%d", &n);
   newNum = (1 << n) | num;
   printf("Bit set successfully.\n\n");
   printf("Number before setting %d bit: %d (in decimal)\n", n, num);
   printf("Number after setting %d bit: %d (in decimal)\n", n, newNum);
   return 0;
}
```

5.

```
#include <stdio.h>
int main()
{
   int num, n, newNum;
   printf("Enter any number: ");
   scanf("%d", &num);
   printf("Enter nth bit to clear (0-31): ");
   scanf("%d", &n);
   newNum = (~(1 << n)) & num;
   printf("Bit cleared successfully.\n\n");
   printf("Number before clearing %d bit: %d (in decimal)\n", n, num);
   printf("Number after clearing %d bit: %d (in decimal)\n", n, newNum);
   return 0;
}
```

6.

```
#include <stdio.h>
int main()
{int num, n, newNum;
   printf("Enter any number: ");
   scanf("%d", &num);
   printf("Enter nth bit to toggle (0-31): ");
   scanf("%d", &n);
   newNum = num ^ (1 << n);
   printf("Bit toggled successfully.\n\n");
   printf("Number before toggling %d bit: %d (in decimal)\n", n, num);
   printf("Number after toggling %d bit: %d (in decimal)\n", n, newNum);
   return 0;}
```

7.

```
#include <stdio.h>
#define INT_SIZE sizeof(int) * 8
int main()
{
   int num, order = -1, i;
   printf("Enter any number: ");
   scanf("%d", &num);
   for(i=0; i<INT_SIZE; i++)
   {
      if((num>>i) & 1)
         order = i;
   }
   if (order != -1)
      printf("Highest order set bit in %d is %d",
num, order);
   else
      printf("0 has no set bits.");
   return 0;
}
```

8.

```
#include <stdio.h>
#define INT_SIZE sizeof(int) * 8 /* Integer size
in bits */
int main()
{
   int num, order, i;
   printf("Enter any number: ");
   scanf("%d", &num);
   order = INT_SIZE - 1;
   for(i=0; i<INT_SIZE; i++)
   {
      if((num>>i) & 1)
      {
         order = i;
         break;
      }
   }

   printf("Lowest order set bit in %d is %d",
num, order);

   return 0;
}
```

9.

```
#include <stdio.h>
#define INT_SIZE sizeof(int) * 8
int main()
{
   int num, count, i;
   printf("Enter any number: ");
   scanf("%d", &num);
   count = 0;
   for(i=0; i<INT_SIZE; i++)
   {
      if((num >> i ) & 1)
      {
         break;
      }
      count++;
   }
   printf("Total number of trailing zeros in %d is
%d.", num, count);
   return 0;
}
```

10.

```
#include <stdio.h>
#define INT_SIZE sizeof(int) * 8
int main()
{
   int num, count, msb, i;
   printf("Enter any number: ");
   scanf("%d", &num);
   msb = 1 << (INT_SIZE - 1);
   count = 0;
   for(i=0; i<INT_SIZE; i++)
   {
      if((num << i) & msb)
      {
         break;
      }

      count++;
   }
   printf("Total number of leading zeros in %d is
%d", num, count);
   return 0;
}
```

11.

```
#include <stdio.h>
int main()
{
   int num, flippedNumber;
   printf("Enter any number: ");
   scanf("%d", &num);
   flippedNumber = ~num;
   printf("Original number = %d (in decimal)\n", num);
   printf("Number after bits are flipped = %d (in decimal)", flippedNumber);
   return 0;
}
```

12.

```
#include <stdio.h>
#define INT_SIZE sizeof(int) * 8 /* Total number of bits in integer */
int main()
{
   int num, zeros, ones, i;
   printf("Enter any number: ");
   scanf("%d", &num);
   zeros = 0;
   ones = 0;
   for(i=0; i<INT_SIZE; i++)
   {
      if(num & 1)
         ones++;
      else
         zeros++;
      num >>= 1;
   }
   printf("Total zero bit is %d\n", zeros);
   printf("Total one bit is %d", ones);
   return 0;
}
```

13.

```
#include <stdio.h>
#define INT_SIZE sizeof(int)
#define INT_BITS INT_SIZE * 8 - 1
int rotateLeft(int num, unsigned int rotation);
int rotateRight(int num, unsigned int rotation);
int main()
{int num;
   unsigned int rotation;
   printf("Enter a number: ");
   scanf("%d", &num);
   printf("Enter number of rotation: ");
   scanf("%u", &rotation);
   printf("%d left rotated %u times = %d\n\n", num, rotation, rotateLeft(num, rotation));
   printf("%d right rotated %u times = %d\n", num, rotation, rotateRight(num, rotation));
   return 0;
}
int rotateLeft(int num, unsigned int rotation)
{
   int DROPPED_MSB;
   rotation %= INT_BITS;
   while(rotation--)
   {
      DROPPED_MSB = (num >> INT_BITS) & 1;
      num = (num << 1) | DROPPED_MSB;
   }
```

```
 return num;
}
int rotateRight(int num, unsigned int rotation)
{
   int DROPPED_LSB;
   rotation %= INT_BITS;
   while(rotation--)
   {
      DROPPED_LSB = num & 1;
      num = (num >> 1) & (~(1 << INT_BITS));
      num = num | (DROPPED_LSB << INT_BITS);
   }
   return num;
}
```

14.

```
#include <stdio.h>
#define INT_SIZE sizeof(int) * 8
int main()
{
   int num, index, i;
   int bin[INT_SIZE];
   printf("Enter any number: ");
   scanf("%d", &num);

   index = INT_SIZE - 1;

   while(index >= 0)
   {
      bin[index] = num & 1;
      index--;
      num >>= 1;
   }
   printf("Converted binary: ");
   for(i=0; i<INT_SIZE; i++)
   {
      printf("%d", bin[i]);
   }

   return 0;
}
```

15.

```
#include <stdio.h>
int main()
{
   int num1, num2;
   printf("Enter any two numbers: ");
   scanf("%d%d", &num1, &num2);
   printf("Original value of num1 = %d\n",
num1);
   printf("Original value of num2 = %d\n",
num2);
   num1 ^= num2;
   num2 ^= num1;
   num1 ^= num2;
   printf("Num1 after swapping = %d\n",
num1);
   printf("Num2 after swapping = %d\n",
num2);
   return 0;
}
```

16.

```
#include <stdio.h>
int main()
{
   int num;
   printf("Enter any number: ");
   scanf("%d", &num);
   if(num & 1)
   {
      printf("%d is odd.", num);
   }
   else
   {
      printf("%d is even.", num);
   }
   return 0;
}
```