1- Write a program that takes an integer input from the user, performs a bitwise NOT operation on it, and then prints the result.

اكتب برنامجًا يأخذ مدخلات صحيحة من المستخدم، وينفذ عملية NOT عليها، ثم يطبع النتيجة.

Input

```
Enter an integer: 5
```

Output

```
Result of bitwise NOT operation: -6
```

Solution

```c
//www.gammal.tech

#include <stdio.h>

int main() {
    int x;

    // Input: Get an integer from the user
    printf("Enter an integer: ");
    scanf("%d", &x);

    // Bitwise NOT operation
    int result = ~x;

    // Output: Display the result
    printf("Result of bitwise NOT operation: %d\n", result);

    return 0;
}
```

2- Write a program that takes two integer inputs from the user, performs a bitwise AND operation on them, and then prints the result.

اكتب برنامجًا يأخذ مدخلين صحيحين من المستخدم، ويجري عملية AND عليهما، ثم يطبع النتيجة.

Input

```
Enter the first integer: 9
Enter the second integer: 7
```

Output

```
Result of bitwise AND operation: 1
```

Solution

```c
//www.gammal.tech

#include <stdio.h>

int main() {
    int x, y;

    // Input: Get two integers from the user
    printf("Enter the first integer: ");
    scanf("%d", &x);

    printf("Enter the second integer: ");
    scanf("%d", &y);

    // Bitwise AND operation
    int result = x & y;

    // Output: Display the result
    printf("Result of bitwise AND operation: %d\n", result);

    return 0;
}
```

3- Write a program that takes two integer inputs from the user, performs a bitwise OR operation on them, and then prints the result.

اكتب برنامجًا يأخذ مدخلين صحيحين من المستخدم، وينفذ عملية bitwise OR عليهما، ثم يطبع النتيجة.

Input

```
Enter the first integer: 5
Enter the second integer: 6
```

Output

```
Result of bitwise OR operation: 7
```

Solution

```c
//www.gammal.tech

#include <stdio.h>

int main() {
    int x, y;

    // Input: Get two integers from the user
    printf("Enter the first integer: ");
    scanf("%d", &x);

    printf("Enter the second integer: ");
    scanf("%d", &y);

    // Bitwise OR operation
    int result = x | y;

    // Output: Display the result
    printf("Result of bitwise OR operation: %d\n", result);

    return 0;
}
```

4- Write a program that takes two integer inputs from the user, performs a bitwise XOR operation on them, and then prints the result.

اكتب برنامجًا يأخذ مدخلين صحيحين من المستخدم، وينفذ عملية XOR عليهما، ثم يطبع النتيجة.

Input

```
Enter the first integer: 5
Enter the second integer: 7
```

Output

```
Result of bitwise XOR operation: 2
```

Solution

```c
//www.gammal.tech

#include <stdio.h>

int main() {
    int x, y;

    // Input: Get two integers from the user
    printf("Enter the first integer: ");
    scanf("%d", &x);

    printf("Enter the second integer: ");
    scanf("%d", &y);

    // Bitwise XOR operation
    int result = x ^ y;

    // Output: Display the result
    printf("Result of bitwise XOR operation: %d\n", result);

    return 0;
}
```

5- Write a program that takes an integer input from the user, performs a left shift operation on it by a specified number of positions, and then prints the result.

اكتب برنامجًا يأخذ مدخلات صحيحة من المستخدم، ويجري عليها عملية إزاحة لليسار بعدد محدد من المواضع، ثم يطبع النتيجة.

Input

```
Enter an integer: 5
Enter the number of positions to left shift: 2
```

Output

```
Result of left shift operation: 20
```

Solution

```c
//www.gammal.tech

#include <stdio.h>

int main() {
    int x, y;

    // Input: Get an integer and the number of positions from the user
    printf("Enter an integer: ");
    scanf("%d", &x);

    printf("Enter the number of positions to left shift: ");
    scanf("%d", &y);

    // Left shift operation
    int result = x << y;

    // Output: Display the result
    printf("Result of left shift operation: %d\n", result);

    return 0;
}
```

6- Write a program that takes an integer input from the user, performs a right shift operation on it by a specified number of positions, and then prints the result.

اكتب برنامجًا يأخذ مدخلات عددية من المستخدم، ويجري عليها عملية إزاحة لليمين بعدد محدد من المواضع، ثم يطبع النتيجة.

Input

```
Enter an integer: 10
Enter the number of positions to right shift: 1
```

Output

```
Result of right shift operation: 5
```

Solution

```c
//www.gammal.tech

#include <stdio.h>

int main() {
    int x, y;

    // Input: Get an integer and the number of positions from the user
    printf("Enter an integer: ");
    scanf("%d", &x);

    printf("Enter the number of positions to right shift: ");
    scanf("%d", &y);

    // Right shift operation
    int result = x >> y;

    // Output: Display the result
    printf("Result of right shift operation: %d\n", result);

    return 0;
}
```

7- Write a program that takes an integer input from the user, performs multiple bitwise NOT operations on it, and prints the result.

اكتب برنامجًا يأخذ مدخلات صحيحة من المستخدم، وينفذ عمليات NOT متعددة عليه، ويطبع النتيجة.

Input

```
Enter an integer: 30
```

Output

```
Result 1: -31
Result 2: 30
Result 3: -31
```

Solution

```c
//www.gammal.tech

#include <stdio.h>

int main() {
    int x;

    // Input: Get an integer from the user
    printf("Enter an integer: ");
    scanf("%d", &x);

    // Perform three bitwise NOT operations
    int result1 = ~x;
    int result2 = ~~x;  // Two consecutive NOT operations cancel each other
    int result3 = ~~~x; // Three consecutive NOT operations cancel each other

    // Output: Display the results
    printf("Result 1: %d\n", result1);
    printf("Result 2: %d\n", result2);
    printf("Result 3: %d\n", result3);

    return 0;
}
```

8- Write a program that takes an array of integers, performs a bitwise NOT operation on each element, and then prints the results.

اكتب برنامجًا يأخذ array من الأعداد الصحيحة، وينفذ عملية NOT على كل عنصر، ثم يطبع النتائج.

Output

```
Updated array after bitwise NOT operation:
-6 -13 -9 -26 -11
```

Solution

```c
//www.gammal.tech

#include <stdio.h>

int main() {
    // Define an array of integers
    int numbers[] = {5, 12, 8, 25, 10};

    // Calculate the size of the array
    int size = sizeof(numbers) / sizeof(numbers[0]);

    // Perform bitwise NOT operation on each element
    for (int i = 0; i < size; i++) {
        numbers[i] = ~numbers[i];
    }

    // Output: Display the updated array
    printf("Updated array after bitwise NOT operation:\n");
    for (int i = 0; i < size; i++) {
        printf("%d ", numbers[i]);
    }
    printf("\n");

    return 0;
}
```

9- Write a program that calculates the bitwise NOT of the smallest negative integer and prints the result.

اكتب برنامجًا يحسب عدد البتات NOT لأصغر عدد صحيح سالب ويطبع النتيجة.

Output

```
Bitwise NOT of the smallest negative integer: 2147483647
```

Solution

```c
//www.gammal.tech

#include <stdio.h>

int main() {
    int x = 1 << 31;

    // Calculate bitwise NOT
    int result = ~x;

    // Output: Display the result
    printf("Bitwise NOT of the smallest negative integer: %d\n", result);

    return 0;
}
```

10- Write a program that takes an integer as input, performs a right shift operation, and then applies the bitwise NOT operation on the result.

اكتب برنامجًا يأخذ عددًا صحيحًا كمدخل، وينفذ عملية إزاحة لليمين، ثم يطبق عملية NOT على النتيجة.

Input

```
Enter a number: 5
```

Output

```
Bitwise NOT of right shifted result: -3
```

Solution

```c
//www.gammal.tech

#include <stdio.h>

int main() {
    int x;

    // Input: Get a number from the user
    printf("Enter a number: ");
    scanf("%d", &x);

    // Right shift operation by 1
    int shiftedResult = x >> 1;

    // Bitwise NOT operation
    int result = ~shiftedResult;

    // Output: Display the result
    printf("Bitwise NOT of right shifted result: %d\n", result);

    return 0;
}
```
