

```
1  #include <stdio.h>
2
3  int main(void)
4  {
5      //function prototypes
6      void PrintBinaryFormOfNumber(unsigned int);
7
8      //variable declarations
9      unsigned int a;
10     unsigned int b;
11     unsigned int result;
12
13     //code
14     printf("\n\n");
15     printf("Enter An Integer = ");
16     scanf("%u", &a);
17
18     printf("\n\n");
19     printf("Enter Another Integer = ");
20     scanf("%u", &b);
21
22     printf("\n\n\n\n");
23     result = a & b;
24     printf("Bitwise AND-ing Of \nA = %d (Decimal) and B = %d (Decimal) gives    ↗
        result %d (Decimal).\n\n", a, b, result);
25
26     PrintBinaryFormOfNumber(a);
27     PrintBinaryFormOfNumber(b);
28     PrintBinaryFormOfNumber(result);
29
30     return(0);
31 }
32
33
34 // ***** BEGINNERS TO C PROGRAMMING LANGUAGE : PLEASE IGNORE THE CODE OF THE    ↗
    FOLLOWING FUNCTION SNIPPET 'PrintBinaryFormOfNumber()' *****
35 // ***** YOU MAY COME BACK TO THIS CODE AND WILL UNDERSTAND IT MUCH BETTER AFTER    ↗
    YOU HAVE COVERED : ARRAYS, LOOPS AND FUNCTIONS *****
36 // ***** THE ONLY OBJECTIVE OF WRITING THIS FUNCTION WAS TO OBTAIN THE BINARY    ↗
    REPRESENTATION OF DECIMAL INTEGERS SO THAT BIT-WISE AND-ing, OR-ing, COMPLEMENT    ↗
    AND BIT-SHIFTING COULD BE UNDERSTOOD WITH GREAT EASE *****
37
38 void PrintBinaryFormOfNumber(unsigned int decimal_number)
39 {
40     //variable declarations
41     unsigned int quotient, remainder;
42     unsigned int num;
43     unsigned int binary_array[8];
44     int i;
45
46     //code
47     for (i = 0; i < 8; i++)
```

```
48     binary_array[i] = 0;
49
50     printf("The Binary Form Of The Decimal Integer %d Is\t=\t", decimal_number);
51     num = decimal_number;
52     i = 7;
53     while (num != 0)
54     {
55         quotient = num / 2;
56         remainder = num % 2;
57         binary_array[i] = remainder;
58         num = quotient;
59         i--;
60     }
61
62     for (i = 0; i < 8; i++)
63         printf("%u", binary_array[i]);
64
65     printf("\n\n");
66 }
67
```