# Bitwise Operators

```
01 #include <stdio.h>
02
03 int main() {
04    int i = 0x7DBEA395;          // 0111 1101 1011 1110 1010 0011 1001 0101
05    int j = 0x66F028C9;          // 0110 0110 1111 0000 0010 1000 1100 1001
06    int k = 0x80000000, mask, bit1, bit2, lsbCount;
07    unsigned short us = 0xB5CD;
08
09    printf("i&j: %8X  i|j: %8X  i^j: %8X  ~i: %8X\n\n", i&j, i|j, i^j, ~i);
10
11    printf("i&0x7F0: %08X  i|0x7F0: %08X  i&~0x7F0: %08X\n\n",
12     i&0x7F0, i|0x7F0, i&~0x7F0);
13
14    printf("i<<4: %08X  i>>6: %08X  k>>24: %08X  i<<32: %08X\n\n",
15     i<<4, i>>6, k>>24, i<<32);
16
17    us = us << 8 | us >> 8;
18    printf("us swapped: %4X\n\n", us);
19
20    printf("Enter bit to set and to clear: ");
21    scanf("%d%d", &bit1, &bit2);
22    printf("%08X\n\n", (i | 1 << bit1) & ~(1 << bit2));
23
24    printf("Keep how many LSBs? ");
25    scanf("%d", &lsbCount);
26    printf("%08X\n", j & (1 << lsbCount) - 1);
27 }
28
29 /* Sample run:
30 i&j: 64B02081  i|j: 7FFEABDD  i^j: 1B4E8B5C  ~i: 82415C6A
31
32 i&0x7F0: 00000390  i|0x7F0: 7DBEA7F5  i&~0x7F0: 7DBEA005
33
34 i<<4: DBEA3950  i>>6: 01F6FA8E  k>>24: FFFFFF80  i<<32: 7DBEA395
35
36 us swapped: CDB5
37
38 Enter bit to set and to clear: 6 8
39 7DBEA2D5
40
41 Keep how many LSBs? 7
42 00000049
43 */
```

### *Bitwise operator examples*

```
i     0111 1101 1011 1110 1010 0011 1001 0101
j     0110 0110 1111 0000 0010 1000 1100 1001
-----------------------------------------------
i&j   0110 0100 1011 0000 0010 0000 1000 0001
i|j   0111 1111 1111 1110 1010 1011 1101 1101
i^j   0001 1011 0100 1110 1000 1011 0101 1100
~i    1000 0010 0100 0001 0101 1100 0110 1010
i<<4  1101 1011 1110 1010 0011 1001 0101 0000
i>>6  0000 0001 1111 0110 1111 1010 1000 1110
```