

Clicker Questions

In case you've forgotten:

AND: this is a single ampersand:	&
OR: this is a single vertical bar:	
XOR: this is a single carat:	^
NOT: this is a single tilde:	~
Left-shift: this is two less-than signs:	<<
Right-shift: this is two greater-than signs:	>>

For i between 1 and 6, Question i is to tell me what gets printed on line i of the program to the right.

```
#include <iostream>
#include <cstdio>
using namespace std;

int main()
{
    int a, b, c, d, e;

    a = 32;
    b = 10;
    c = 0x3b53a;
    d = 0x3b93a;
    e = 0x8372a;

    printf("%d\n", 5 & 9);           // Question 1
    printf("%d\n", 3 | 12);         // Question 2
    printf("0x%x\n", a + b);        // Question 3
    printf("0x%x\n", c ^ d);        // Question 4
    printf("0x%x\n", e << 8);       // Question 5
    printf("0x%x\n", e >> 4);       // Question 6
    return 0;
}
```

Clicker Question Answers

Here's the short version

```
UNIX> g++ q1.cpp
UNIX> a.out
1
15
0x2a
0xc00
0x8372a00
0x8372
UNIX>
```

Here's the longer version

Question 1:

```
0101 -- 5 in binary
1001 -- 9 in binary
----
0001 -- so the answer is 1
```

Question 2:

```
0011 -- 3 in binary
1101 -- 12 in binary
----
1111 -- so the answer is 15
```

Question 3: 32 is 2 times 16, so the hex digit in the 16's place is 2. 10 is 0xa in hex. So the answer is 0x2a.

Question 4: In hex, each digit is four bits, so you can simply do the xor in hex:

```
3 b 5 3 a
3 b 9 3 a
-----
0 0 c 0 0 -- the answer is 0xc00
```

Question 5: Remember that left and right shifting by multiples of 4 allow you to simply work on the hex digits. So you simply append two zeros to the hex: 0x8372a00.

Question 6: And here you simply chop off one digit: 0x8372.