Lab #7

Exercise #1: Printing Binary Representation

Write an assembly program which repeatedly prompts the user to enter a signed integer (recall that `read_int` reads in 4-byte values into `eax`). The program stops immediately if the integer is equal to zero. For each entered positive integer the program prints out the **binary** representation of the number. You can assume that the user enters valid numbers. Here is an example interaction with the program, with user input in boldface:

```
Enter an integer: 129
binary representation: 00000000000000000000000010000001
Enter an integer: -88
binary representation: 11111111111111111111111110101000
Enter an integer: 142341
binary representation: 00000000000000100010110000000101
Enter an integer: -591023
binary representation: 11111111111101101111101101010001
Enter an integer: 0
```

Exercise #2: Printing Hex Representations

Write an assembly program which repeatedly prompts the user to enter a signed integer (recall that `read_int` reads in 4-byte values into `eax`). The program stops immediately if the integer is equal to zero. For each entered positive integer the program prints out the **hex** representation of the number. You can assume that the user enters valid numbers. Here is an example interaction with the program, with user input in boldface:

```
Enter an integer: -1
hex representation: FFFFFFFF
Enter an integer: 1045
hex representation: 00000415
Enter an integer: -41251
hex representation: FFFF5EDD
Enter an integer: 51231
hex representation: 0000C81F
Enter an integer: 0
```