Computer Science 240
Binary Operations
Assignment for Lab 3

Perform addition on the following binary and hexadecimal numbers (assume **two's complement** format!). Indicate whether there is a carry-out or an overflow for each addition.

For the first 2 calculations, assume **16- bit representation.** Do the calculation using the binary values.

Then, convert the numbers for the operands and result to hexadecimal notation (to convert, divide the digits into groups of 4, and translate each group to the corresponding hexadecimal value).

Carry-Out? Overflow? Hexadecimal Value

1. 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
   + 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

2. 0 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0
   + 0 1 1 1 1 1 1 1 1 0 0 1 1 0 0 1

Now, assume **32-bit representation,** using hexadecimal notation.

Carry-Out? Overflow? ____

3. 0x A A F F 9 0 1 4
   + 0x A A E 3 C D 1 2

4. 0x 7 F A A 3 2 7 8
   + 0x 6 0 2 4 C D 1 2

The following short C program accepts two integer (32-bit) hexadecimal numbers from the user as inputs and prints the sum, including the overflow and carryout:

```
#include <stdio.h>
#include <stdlib.h>

void add(int x,int y) {
  int sum = x+y;
  printf("\n\nx + y  = 0x%x (decimal %d)",sum,sum);
```

```
  int overflow = ((x > 0 && y > 0 && sum < 0)|| (x < 0 && y < 0 && sum > 0));
  printf("\nOverflow = %d",overflow);

  long int carryout = (((long int)x + (long int)y) >> 32) & 1;
  printf("\nCarryout = %ld\n",carryout);
}

int main()
{
   int x;
  int y;
  printf("Enter a hex value for x: 0x");
  scanf("%x",&x);
  printf("(decimal %d)",x);
  printf("\nEnter a hex value for y: 0x");
  scanf("%x",&y);
  printf("(decimal %d)",y);

  add(x,y);
  return 0;
}
```

In a Terminal, use emacs to create a file *sum.c* containing this program. Compile and run the program for the values given above. Verify your results.

5. Explain the code for generating the overflow and carryout:

Run the program with the following inputs:

0x 7 F A A 3 2 7 8
0x 6 0 2 4 C D 1 2

Describe the result: