CISC 260 Machine Organization and Assembly Language

Assignment # 1 Solution

1. Number conversion:

Decimal	Hex	Binary
-35	0xDD	1101 1101
-52	0xCC	1100 1100
73	0x49	0100 1001
58	0x3A	0011 1010
-25	0xE7	1110 0111

2. ASCII code conversion:
   a. X490341524D
      I♥ARM
   b. CS=Fun
      0x43533D465564E

3. $x = 0101\ 0101_{two}$ and $y = 1100\ 0101_{two}$
   $x_{10} = 85$
   $y_{10} = -59$
   a. x + y
      Expected answer: 85 + (-59) = 26 ~ 0001 1010
      Actual answer:

```
  0101 0101
 +1100 0101
-----------
1    0001 1010 ~ 26
```

      Overflow: No

   b. x – y
      Expected answer: 85 – (-59) = 144 ~ 1001 0000
      Actual answer:

```
  0101 0101
 +0011 1010
-----------
 1001 0000 ~ 144
```

      Overflow: Yes

4. Booth's algorithm:

```
#include <stdio.h>


int multBooth (int q, int m){
  int a = 0;
  int q_neg1 = 0;
  int q_0 = 0;
```

```
  int i = 16;
  while (i > 0)
    {
      q_0 = q & 1;
      if (q_0 == 1 && q_neg1 == 0) // 10
     {
       a -= m;
     }
      else if (q_0 == 0 && q_neg1 == 1) // 01
     {
       a += m;
     }
      q_neg1 = q_0;
      m <<= 1;
      q >>= 1;
      i -= 1;
    }
  return a;
}

void main (){
  int q, m, a;
  printf ("Enter an integer:\n");
  scanf ("%d", &q);
  printf ("Enter an integer:\n");
  scanf ("%d", &m);

  // the code of your subroutine multBooth is called below
  a = multBooth (q, m);
  printf ("the product = %d\n", a);
}
```