Name: _____

Directions: **Work only on this sheet** (on both sides, if needed); do not turn in any supplementary sheets of paper. There is actually plenty of room for your answers, as long as you organize yourself BEFORE starting writing.

1. (20) Consider the output of the code on p.29. Suppose we had a 64-bit system. Assuming the address of **X** did not change, what would be the address of **Y**?

2. (20) Consider the declaration

```
char q[8][3];
```

If **q[0][2]** happens to be stored at address 0x20c, at what address will **q[2][1]** be stored?

3. (15) Consider the code

```
unsigned int *f,*g; char *s1 = "gr88", *s2 = "bb99";
f = s1; g = s2;
printf("%u\n",f-g);
```

Fill in the blank: The number printed out will be negative if and only if the machine _____.

4. The function **print5()** below prints out an **unsigned int** in base-5. So,

```
unsigned int x = 39;
print5(x);
```

would output 124 (actually 0124, since leading 0s aren't suppressed here). Assume we have 8-bit words.

(a) (30) Fill in the blanks in the code:

```
int print5(unsigned int x) {
   unsigned int i,d, power5=_____, r=x;
   for (i = 0; i < 4; i++) {
      d = r / power5;
      r = r % power5;
      printf(_____);
      power5 /= 5;
   }
   printf("\n");
}
```

(b) (15) What impact, if any, did the assumption of an 8-bit word size have on the code?

Solutions:

1. 0xbffffb7c

2 0x211

3. is little-endian

4a

```
int print5(int x) {
   int i,d, power5=125, r=x;
   for (i = 0; i < 4; i++) {
      d = r / power5;
      r = r % power5;
      printf("%c",d+48);
      power5 /= 5;
   }
   printf("\n");
}
```

4b the initial value of `power5` was 125