1. `sort words | grep q | head -1`

2. ```c
   double sumArray(double *values, int values_len) {
     double total;
     int i;
     total = 0.0;
     for (i = 0; i < values_len; i++) {
       total += values[i];
     }
     return total;
   }
``` 

3. ```
   5 7
   7 7
``` 

4. \(32 \times 2^{12}\)

5. a. \((-14)^3 = -13\)
    b. \((-10) \gg 2 = -3\)

6. ```c
   int mult80(int n) {
     return (n << 6) + (n << 4);
   }
``` 

7. a. \(0 0011 000\)
    b. \(-4.25_{10}\)
    c. \(7/16\)
    d. \(00000001\), which converts to \(1/64\)
    e. \(01101111\), which converts to \(15.5_{10}\)

8. Suppose \(x = 104, y = -104, \text{ and } z = 1\). Then

\[
(x + y) + z = (104 + (-104)) + 1 = 0 + 1 = 1
\]

but

\[
(x + z) + y = (104 + 1) + (-104) = 104 + (-104) = 0.
\]

(We say \(104 + 1 = 104\) because 105 is not representable as an eight-bit floating-point number, and so we must round to 104.)