Charlie Melnarile

1 String a = new String ("Hunter");
   String b = ("hunter");

2 byte a = 68;
   byte b = 12;

3 If you have a + b + c and a + b are
   bigger than the machine can handle but c
   is a negative number that brings the equation
   under the max int size, it can still be computed,
   if the second addition is done first.

4 A: $((((a * (b-1)^1)^2 / (c))^3 \bmod d)^4$

   B: $((-a)^1 \text{ or } ((c = d)^2 \text{ and } e)^3)^4$

   C: $(-(a + b)^1)^2$

5 ⟨expression⟩ → ⟨expression⟩ or ⟨or exp⟩
                   | ⟨expression⟩ xor ⟨or exp⟩
   ⟨and exp⟩ → ⟨and exp⟩ and ⟨and exp⟩

   ⟨equal exp⟩ → ⟨equal exp⟩ = ⟨equal exp⟩
                   | ⟨equal exp⟩ /= ⟨equal exp⟩
                   | ⟨equal exp⟩ < ⟨equal exp⟩
                   | ⟨equal exp⟩ <= ⟨equal exp⟩
                   | ⟨equal exp⟩ >= ⟨equal exp⟩
                   | ⟨equal exp⟩ > ⟨equal exp⟩
   ⟨unary exp⟩ → ⟨unary exp⟩ -(unary) ⟨unary exp⟩
   ⟨add exp⟩ → ⟨add exp⟩ + ⟨add exp⟩
                   | ⟨add exp⟩ - ⟨add exp⟩
                   | ⟨add exp⟩ & ⟨add exp⟩
                   | ⟨add exp⟩ mod ⟨add exp⟩
   ⟨mult exp⟩ → ⟨mult exp⟩ * ⟨mult exp⟩
                   | ⟨mult exp⟩ / ⟨mult exp⟩
                   | ⟨mult exp⟩ not ⟨mult exp⟩
   ⟨operand⟩ → a | b | c | d | e