Problem 1:

1. One's complement of a F
2. a B
3. a & b A
4. a * 7 C
5. a / 4 E
6. (a < 0) ? 1 : -1 H

Problem 2:

Expression	Always True?
(x < y) == (-x > -y)	N
((x + y) << 4) + y - x == 17 * y + 15 * x	Y
~x + ~y + 1 == ~(x + y)	Y
ux - uy == -(y - x)	Y
(x >= 0) || (x < ux)	N
((x >> 1) << 1) <= x	Y
(double)(float) x == (double) x	N
dx + dy == (double) (y + x)	N
dx + dy + dz == dz + dy + dx	Y
dx * dy * dz == dz * dy * dx	N

Problem 3:

Number	Decimal Representation	Binary Representation
Zero	0	00 0000
n/a	-1	11 1111
n/a	5	00 0101
n/a	-10	11 0110
n/a	26	01 1010
n/a	-26	10 0110

TMax	31	01 1111
TMin	-32	10 0000
TMax+TMax	-2	11 1110
TMin+TMin	0	00 0000
TMin+1	-31	10 0001
TMin−1	31	01 1111
TMax+1	-32	10 0000
−TMax	-31	10 0001
−TMin	-32	10 0000

Problem 4:

Description	Hex	M	E	V
-0	8000	0	-62	-
Smallest value>2	4001	257/256	1	257/128
512	4800	1	9	-
Largest denormalized	00FF	255/256	-62	255×2^{-70}
-∞	FF00	-	-	-
Number with hex representation 3BB0	-	27/16	-4	27/256

Problem 5:

A		B	
Bits	**Value**	**Bits**	**Value**
1 01111 001	**-9/8**	**1 0111 0010**	**-9/8**
0 10110 011	176	0 1110 0110	176
1 00111 010	-5/1024	1 0000 0101	-5/1024

0 00000 111	7/131072	0 0000 0001	1/1024
1 11100 000	-8192	1 1110 1111	-248
0 10111 100	384	0 1111 0000	+∞

Problem 6:

```
float fpwr2(int x)
{
      /* Result exponent and fraction */
      unsigned exp, frac;
      unsigned u;

      if (x < -149) {
            /* Too small. Return 0.0 */
            exp = 0;
            frac = 0;
      } else if (x < -126) {
            /* Denormalized result */
            exp = 0;
            frac = 1 << (x + 149);
      } else if (x < 128) {
            /* Normalized result */
            exp = x + 127;
            frac = 0;
      } else {
            /* Too big. Return +oo*/
            exp = 255;
            frac = 0;
      }

      /* Pack exp and frac into 32 bits */
      u = exp << 23 | frac;
      /* Return as float */
      return u2f(u);
}
```