```
# include <iostream>
# include <vector>
# include <cmath>
# include <iomanip>
using namespace std;

// A container to hold a a float and a double value.
struct FloatDoublePair {
    float f = 0;
    double d = 0;
};

// A container to hold a an int and a long value.
struct IntLongPair {
    int i = 0;
    long l = 0;
};

FloatDoublePair p1() {
    // Variables for the machine precision.
    float eps_f = 1.0f;
    float prev_eps_f = 0.0f;
    double eps_d = 1.0;
    double prev_eps_d = 0.0;

    // Find precision for floats.
    while(1.0f + eps_f != 1.0f) {
        prev_eps_f = eps_f;
        eps_f /= 2.0f;
    }

    // Find precision for doubles.
    while(1.0 + eps_d != 1.0) {
        prev_eps_d = eps_d;
        eps_d /= 2.0;
    }

    FloatDoublePair output;
    output.f = prev_eps_f;
    output.d = prev_eps_d;

    // Print out.
    cout << scientific << setprecision(10);
    cout << "Float epsilon:  " << prev_eps_f << endl;
    cout << "Double epsilon: " << prev_eps_d << endl;

    return output;
}

IntLongPair p3() {
    //
    int i = 200 * 300 * 400 * 500;
    long l = 200L * 300L * 400L * 500L;
    IntLongPair output;
    output.i = i;
```

```cpp
    output.l = l;

    // Print out.
    cout << scientific << setprecision(10);
    cout << "int value:  " << to_string(i) << endl;
    cout << "long value: " << to_string(l) << endl;

    return output;
}

int p4() {
    unsigned int counter = 0;
    for(int i = 0; i < 3; ++i) --counter;
    cout << "Counter: " << to_string(counter) << endl;
    return counter;
}

int main() {
    FloatDoublePair output_1 = p1();
    IntLongPair output_3 = p3();
    int output_4 = p4();

    return 0;
}
```

1)

$\varepsilon_{float} \approx 1.19 \cdot 10^{-7}$

$\varepsilon_{double} \approx 2.22 \cdot 10^{-16}$

2)

Largest SP $\Big\}$ $V = (-1)^S \cdot 1.F \cdot 2^{E - \overbrace{127}^{b}}$ $\Bigg\}$ $S = 0$
$F = 11\ldots_{23}$
$E = 11\ldots_8 - 1 = 254$
$b = 127$ ↗ no ∞/NaN

$$\rightarrow V_{max} = (-1)^0 \left[1 + \sum_{i=1}^{23} 2^{-i}\right] 2^{254-127}$$

$$= \left[1 + 1 - 2^{-23}\right] 2^{127}$$

$$= \boxed{2^{128} - 2^{104}}$$

(most negative)
Smallest SP $\Big\}$ $S = 1$ $\quad E = 254$
$F = 11\ldots_{23}$

$$\rightarrow V_{min} = -V_{max} = \boxed{2^{104} - 2^{128}}$$

Largest DP $\Big\}$ $S = 0$ $\quad E = 11\ldots_{11} - 1 = 2046$
$F = 11\ldots_{52}$ $\quad b = 1023$ ↗ no ∞/NaN

$$\rightarrow V_{max} = (-1)^0 \left[1 + \sum_{i=1}^{52} 2^{-i}\right] \cdot 2^{2046-1023}$$

$$= \left[2 - 2^{-52}\right] \cdot 2^{1023} = \boxed{2^{1024} - 2^{971}}$$

$$\rightarrow V_{min} = \boxed{2^{971} - 2^{1024}}$$ with $S = 1$ here

3) I get -884901888 because of overflow.

4)

counter = 4294967293

5) <u>SP #</u>

sign — exponent — mantissa
1 bit 8 bits 23 bits

2 2^8 2^{23}

So there are $2 \cdot 2^8 \cdot 2^{23} = \underbrace{2^{32}}$ total SP #'s. To count the non normalized #'s, we have $E = 0$, and must leave out ± 0. So we'd have $2 \cdot 2^{23} - 2 = \underbrace{2^{24} - 2}$ non-normalized numbers, so there must be $\underbrace{2^{32} - (2^{24} - 2)}$ normalized #'s (including ∞'s / NaNs)

6)

normalized : $(-1)^S \; 1.F \cdot 2^{E-b} \rightarrow$ | $b = 2^{k-1} - 1$

$= (-1)^S \; 1.F \cdot 2^{E-3}$ | $= 2^2 - 1 = 3$

un-normalized : $(-1)^S \; 0.F \cdot 2^{-3}$

S	F	E	
0	00	0	4
1	01	1	5
	10	2	6
	11	3	7

a) $E > 0$, $S = 0$

$1.00 \cdot 2^{-2} = 1/4$
$1.00 \cdot 2^{-1} = 1/2$
$1.00 \cdot 2^{0} = 1$
$1.00 \cdot 2^{1} = 2$
$1.00 \cdot 2^{2} = 4$
$1.00 \cdot 2^{3} = 8$
$\cancel{1.00 \cdot 2^{4} = 16}$ ∞

$5/4 \cdot 1/4 = 5/16$
$1.01 \cdot 2^{-2} =$
$1.01 \cdot 2^{-1} = 5/8$
$1.01 \cdot 2^{0} = 5/4$
$1.01 \cdot 2^{1} = 5/2$
$1.01 \cdot 2^{2} = 5$
$1.01 \cdot 2^{3} = 10$
$\cancel{1.01 \cdot 2^{4} = 20}$
NaN

$1 + 1/2$
$1.10 \cdot 2^{-2} = \frac{3}{2} \cdot \frac{1}{4} = \frac{3}{8}$
$1.10 \cdot 2^{-1} = 3/4$
$1.10 \cdot 2^{0} = 3/2$
$1.10 \cdot 2^{1} = 3$
$1.10 \cdot 2^{2} = 6$
$1.10 \cdot 2^{3} = 12$
$\cancel{1.10 \cdot 2^{4} = 24}$
NaN

$1.11 \cdot 2^{-2} = \frac{7}{4} \cdot \frac{1}{4} = \frac{7}{16}$
$1.11 \cdot 2^{-1} = 7/8$
$1.11 \cdot 2^{0} = 7/4$
$1.11 \cdot 2^{1} = 7/2$
$1.11 \cdot 2^{2} = 7$
$1.11 \cdot 2^{3} = 14$
$\cancel{1.11 \cdot 2^{4} = 28}$ NaN

$1.00 \cdot 2^{-3}$
$\parallel$ by convention
$+0$

when $S = 1$, we just get the negatives of the above numbers, including -0

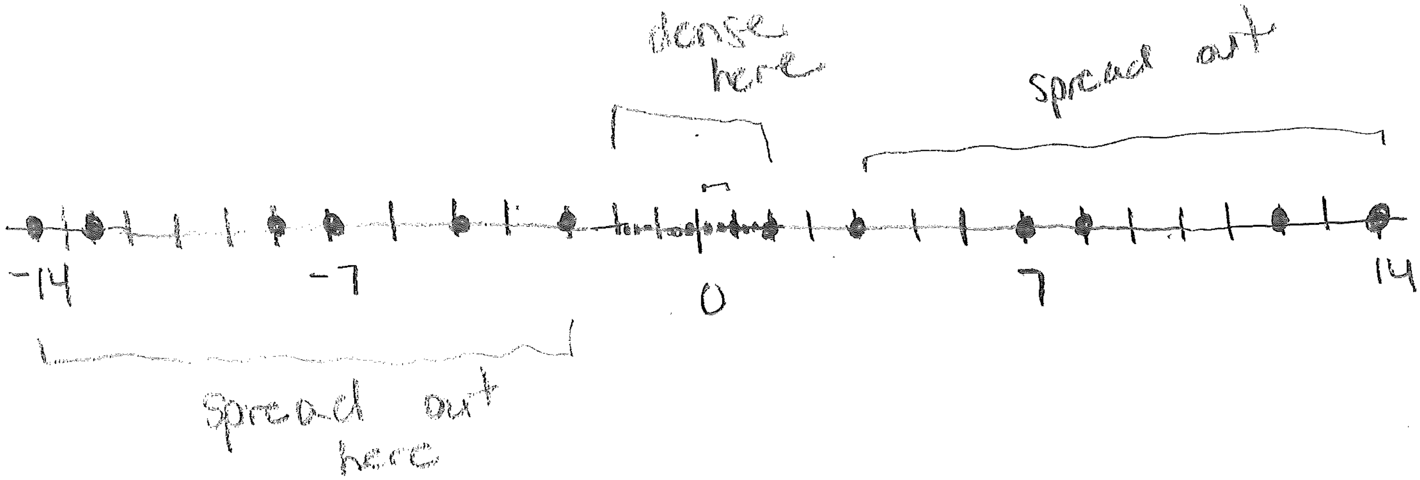
b) $E = 0$, $S = 0$, $F \neq 00$

$0.01 \cdot 2^{-3} = 1/4 \cdot 1/8 = 1/32$
$0.10 \cdot 2^{-3} = 1/2 \cdot 1/8 = 1/16$
$0.11 \cdot 2^{-3} = 3/4 \cdot 1/8 = 3/32$

also the negatives when $S = 1$

$\downarrow$ 6 total

(c)



7)

a) $(D3\overset{11}{B}701)_{16} = 1\cdot 16^0 + \cancel{0\cdot 16^1} + 7\cdot 16^2 + 11\cdot 16^3 + 3\cdot 16^4 + 13\cdot 16^5$

$= 1 + 7\cdot 16^2 + 11\cdot 16^3 + 3\cdot 16^4 + 13\cdot 16^5 = \boxed{13899945}$

b) $(\underbrace{\overset{A}{1010}}\,\underbrace{\overset{1}{0001}}\,\underbrace{\overset{3}{0011}}\,\underbrace{\overset{F}{1111}})_2$

$= \boxed{(A13F)_{16}}$

8)
$$6a + 9b + 15c = 107$$

$$\rightarrow \quad 2a + 3b + 5c = \frac{107}{3}$$

(if $a,b,c \in \mathbb{Z}$)

The LHS is an integer and the RHS is a rational, so a,b,c can't all be integers.

9) Yes, $(\mathbb{Z}_n, +, \cdot)$ is a ring. First we show $(\mathbb{Z}_n, +)$ is an abelian group. To do so, we need for $a, b, c \in \mathbb{Z}_n$

① $(a+b)+c = a+(b+c)$

② $a+b = b+a$

③ $\exists\, z \in \mathbb{Z}_n$ s.t $z + a = a$, call $z := 0$

④ $\exists\, -a \in \mathbb{Z}_n$ s.t $a + (-a) = 0$

For ①, take $a \in [i]$, $b \in [j]$, $c \in [l]$, so that $a = i + K_1 n$, $b = j + K_2 n$, and $c = l + K_3 n$, then

- $(a+b)+c \mod n = [(i+j)+(K_1+K_2)n] + c \mod n$

$= [(i+j+l) + (K_1+K_2+K_3)n \mod n] = i+j+K \mod n$

and

- $a + (b+c) \mod n = a + [(j+l) + (K_2+K_3)n] \mod n$

$= (i+j+l) + (K_1+K_2+K_3)n \mod n$

$= i+j+l \mod n$ ✓

② Follows simply by normal addition being abelian. (very easy) ✓

③ You can take $z = 0$. ✓ $(z \in [0])$

④ If $a \in [i]$, $-a \in [n-i]$ since

$i + K_1 n + (n-i) + K_4 n \mod n = n + (K_1+K_4)n \mod n$

$= 0 \mod n$ ✓

Next, we need $(\mathbb{Z}_n, \cdot)$ to be a monoid.
For this, we need

(A) $(a \cdot b)c = a(b \cdot c)$

(B) $\exists\ 1 \in \mathbb{Z}_n$ s.t $1a = a1 = a$ in $\mathbb{Z}_n$

For (A),

$$\underset{\text{mod } n}{(a \cdot b)c} = ((i+k_1 n)(j+k_2 n))(\overbrace{l+k_3 n}^{c}) \text{ mod } n$$

$$= (ij + (jk_1 + ik_2)n + k_1k_2n^2)(l+k_3 n) \text{ mod } n$$

$$= ijl + \cancel{l(jk_1 + ik_2)n} + \cancel{k_1k_2 l n^2} + \cancel{ijk_3 n} + \cancel{k_3(jk_1 + ik_2)n^2} + \cancel{k_1k_2k_3 n} \text{ mod } n$$

$$= [ijl] \text{ mod } n$$

and

$$\underset{\text{mod } n}{a \cdot (bc)} = [i + k_1 n][(j+k_2 n)(l+k_3 n)] \text{ mod } n$$

$$= [i + k_1 n][jl + (jk_3 + lk_2)n + k_2k_3n^2] \text{ mod } n$$

$$= [ijl + \cancel{i(jk_3 + lk_2)n} + \cancel{ik_2k_3n^2} + \cancel{k_1 jl n} + \cancel{(jk_3 + lk_2)k_1n^2} + \cancel{k_1k_2k_3n^3}] \text{ mod } n$$

$$= [ijl] \text{ mod } n$$

So (A) is ✓.

For (B), we can take $1 \in [1]$, that will work very obviously. So $(\mathbb{Z}_n, \cdot)$ is a monoid. Now, finally, the last thing we need is

$$*_1 \quad a(b+c) = ab + ac$$

$$*_2 \quad (b+c)a = ba + ca$$

For $*_1$,

$$a(b+c) \bmod n = (i+k_1 n)\left[(j+l) + (k_2+k_3)n\right] \bmod n$$

$$= (j+l)i + \cancel{(j+l)k_1 n} + \cancel{i(k_2+k_3)n} + \cancel{k_1(k_2+k_3)n^2} \bmod n$$

$$= (j+l)i \bmod n$$

$$= ij + il \bmod n$$

and

$$ab + ac = (i+k_1 n)(j+k_2 n) + (i+k_1 n)(l+k_3 n) \bmod n$$

$$= ij + \cancel{(ik_2 + jk_1)n} + \cancel{k_1 k_2 n^2} + il + \cancel{(lk_1 + ik_3)n} + \cancel{k_1 k_3 n^2} \bmod n$$

$$= ij + il \bmod n$$

So $*_1$ ✓

For $\star_2$

$$(b+c)a \mod n = \left[(j+\ell)+(K_2+K_3)n\right]\left[i+K_1 n\right] \mod n$$

$$= (j+\ell)i + \cancel{i(K_2+K_3)n} + \cancel{K_1 n(j+\ell)} + \cancel{K_1(K_2+K_3)n^2} \mod n$$

$$= ij + i\ell \mod n$$

and

$$ba + ca \overset{\text{mod } n}{=} (j+K_2 n)(i+K_1 n) + (\ell+K_3 n)(i+K_1 n) \mod n$$

$$= ji + \cancel{(K_2 i + K_1 j)n} + \cancel{K_1 K_2 n^2} + \ell i + \cancel{(\ell K_1 + iK_3)n} + \cancel{K_3 K_1 n^2} \mod n$$

$$= ij + i\ell \mod n, \text{ so } \star_2 \checkmark.$$

Together, this shows that $(\mathbb{Z}_n, +, \cdot)$ is a ring.