Assume the **IEEE single precision** floating-point format ($p = 24$).

1. Determine the representations of 2, 30, 31, $-32$, $20.75 \times 2^{100}$, and 0.2.

2. Determine the representation of $1/10$ and of $1 + 2^{-25}$ using each of the four rounding modes.

3. Recall that the smallest positive normalized number is $N_{\min} = 2^{-126}$ and the largest normalized number is $N_{\max} = (2 - 2^{-23}) \times 2^{127}$. For $x$ in the normalized range $N_{\min} \leq |x| \leq N_{\max}$, the floating-point value $\tilde{x}$ of $x$ satisfies

   $$\tilde{x} = x (1 + \text{error}),$$

   where the magnitude of the error is no greater than $\epsilon = 2^{-23}$, machine epsilon. Does this estimate hold for $0 < x < N_{\min}$?

4. Let $a_1, a_2, \ldots, a_n$ be given floating-point numbers. Consider evaluating the product $a_1 \cdot a_2 \cdot \ldots \cdot a_n$. Follow the same logic as in the example on summation (see A&H 2.4) to estimate the effect of rounding after each multiplication (for simplicity, ignore terms nonlinear in $\epsilon$). How large can the error be? How large can the relative error be? Does the order of multiplication matter?