コンピューターの計算の誤差

0.1 ＋ 0.2 を計算すると、なぜ 0.3 ちょうどにならないことがあるのだろう

10進数でも起こる：1 ÷ 3

1 ÷ 3 ＝ 0.3333…（3がどこまでも続く）

小数第3位までで切ると　→　0.333

0.333 × 3 ＝ 0.999

→ もとの 1 にもどらない（切ったぶんのずれ）

2進数では、0.1 がちょうど表せない

0.1（10進数）＝

0.0 0011 0011 0011 0011 …（2進数）

「0011」がくり返し、終わらない

コンピューターは、決まった桁数で切って記録する

→ 0.1 に近いが、少しずれた値になる

（例：0.1000000000000000055511…）

その結果

0.1 ＋ 0.2 ＝ 0.30000000000000004

プログラムで「0.1 + 0.2 が 0.3 と等しいか」を調べると、「等しくない」になる。

多くのプログラミング言語で、小数を浮動小数点数（倍精度）で計算したときの結果

プログラムで気をつけること

1 等しいかの比べ方

「ちょうど等しいか」ではなく、差がとても小さいかで比べる

2 お金は整数で計算する

円（や銭）を整数にして計算する

3 誤差は積み重なる

計算をくり返すと、ずれが大きくなることがある

4 表せる数にも限りがある

大きすぎる・小さすぎる数は表せない

※ 数値は IEEE 754 の倍精度の浮動小数点数の場合。表示される桁数は言語や設定によってちがう。