

# 本地練習題

5 題 · 本地 gcc/clang · 總分 動手寫

做法：讀題 → 自己寫 → `./run.sh 你的檔.c` 編譯執行 → 對答案看 [參考解答/](#)。用 `scanf` 的程式執行後要自己輸入。

參考解答是可編譯執行的程式檔（[參考解答/01\\_digits.c](#) … [05\\_safe\\_div.c](#)）。注意整數除法、``==`` 與 ``=`、短路求值。

## § 1 練習題

### 練習 1：拆數字

讀入一個四位數整數，用 `/` 和 `%` 分別印出它的千、百、十、個位（空格分隔）。

#### 引導步驟

1. 個位 `n % 10`；十位 `(n / 10) % 10`；百位 `(n / 100) % 10`；千位 `n / 1000`。

#### 易錯

· 用浮點除法（要整數 `/`）。 **參考解答：** [參考解答/01\\_digits.c](#)

## 練習 2：判斷奇偶

讀入一個整數，用 `%` 判斷並印出 `odd` 或 `even`。

### 引導步驟

1. `n % 2 == 0` 為偶數。
2. 可用三元 (cond) ? "even" : "odd" 或 if/else。

### 易錯

- 負數的 `%`：`-7 % 2` 在 C 是 `-1`，判斷奇偶請用 `n % 2 == 0` 較穩。 **參考解答：** [參考解答/02\\_oddeven.c](#)

## 練習 3：秒轉分秒

讀入總秒數，印出「X 分 Y 秒」（用 `/` 取分、`%` 取秒）。

### 引導步驟

1. 分 = `s / 60`；秒 = `s % 60`。

### 易錯

- 把分秒算反；用浮點。 **參考解答：** [參考解答/03\\_sec.c](#)

### 練習 4：正確的相等比較

讀入一個整數 `x`，若它等於 3 印 `yes`，否則印 `no`。特別注意用 `==` 不是 `=`。

#### 引導步驟

1. `if (x == 3)`（兩個等號）。寫成 `=` 會變指派、永遠成立。

#### 易錯

- 把 `==` 寫成 `=`（最經典 bug）。參考解答：[參考解答/04\\_compare.c](#)

### 練習 5：短路求值避免除以零

讀入兩個整數 `a`、`b`。用**短路求值**：`b` 不為 0 才判斷 `a` 是否被 `b` 整除，印出結果；`b` 為 0 時印 `cannot divide by zero`。

#### 引導步驟

1. `if (b != 0 && a % b == 0)` —— `b==0` 時右邊 `a % b` 不會執行（短路）。
2. 否則分別處理 `b==0` 與「不整除」。

#### 易錯

- 把條件順序寫反（`a % b == 0 && b != 0`）會先除以零。參考解答：[參考解答/05\\_safe\\_div.c](#)