

氣泡排序(Bubble Sort)

氣泡排序的主要想法是，透過相鄰的2個值，進行兩兩相比。根據須要設定相比條件會有以下幾個版本。

1. 從後面兩兩比較，小的換到前面 => 「第一輪後，第一個會是最小的值」。✔
2. 從後面兩兩比較，大的換到前面 => 「第一輪後，第一個會是最大的值」。
3. 從前面兩兩比較，小的換到後面 => 「第一輪後，最後一個會是最小的值」。
4. 從前面兩兩比較，大的換到後面 => 「第一輪後，最後一個會是最大的值」。

觀察 (版本1)

Bubble Sort

15 14 4 11 5
1 2 3 4 5

| 第幾輪 | 第1次 | 第2次 | 第3次 | 第4次 | 比較位置 |
|---------|-----|-----|-----|-----|---------|
| 1 | | | | | 5,4,3,2 |
| 2 | | | | | 5,4,3 |
| 3 | | | | | 5,4 |
| 4 | | | | | 5 |
| 1,2,3,4 | | | | | |

TNH 2023年02月15日 16:03

目錄

氣泡排序分析 ① 交換、② 比較 ③ 3張與4張牌 ④ 比幾輪 整合

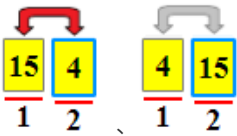
排序過程

模式識別

共比10組，8組有交換，2組沒交換。

- 模式：
 - ① 交換
 - ② 選擇結構(換/不換)。

抽象化



1. 先完成①兩數交換(清單第1，及第2的交換)。
2. 加入②換/不換的條件判斷式。

演算法設計

1. 使用 變數 [] 設為 []、 [] 的第 [] 項、 替換 [] 的第 [] 項為 []、 []
 - ① 變數暫存 設為 紙牌清單第1項。
 - ② 替換紙牌清單第1項 為 紙牌清單第2項。
 - ③ 替換紙牌清單第2項 為 變數暫存
2. 使用 如果 [] 那麼 []
 - ① 換/不換 的條件： 紙牌清單第2項 < 紙牌清單第1項。
 - ② 將 1.的交換程序置於條件判斷式內。
3. 加入比較動畫、交換動畫。



3張牌重複比

排序過程

模式識別

1. 第 2, 3 張比
2. 第 1, 2 張比
3. 第 2, 3 張比

🐞 程式可重複套用 → 把程式改寫成帶參數的函式

1. 使用  定義、 引用。
 - ① 函式命名：比較交換，加入2個參數：前數、後數
 - ② 用前數取代「交換程式」中的1，後數取代「交換程式」中的2。
2. 每次是 第 項，第 項 相比。

抽象化(4張牌重複比)

1. 比較位置 會變 → 比較位置是變數。
2. 怎麼變？
 - ① 比較位置 起始值每輪設為 。(為何?)
 - ② 每重複比一次，比較位置就 。
 - ③ 比較位置 在第1輪： 。
 - 比較位置 在第2輪：、最後1輪：

演算法設計

使用 、、

目錄 ▾ + 📄 ⚙️ 📄 📄 ▾ <<

重複 次 — 想想看，執行多少次要如何算？

... — 要重複執行的程式塊

說出 字串組合 比較位置 - 字串組合 , 比較位置 持續 秒 — 測試用，測試好拿掉

變數 比較位置 ▾ 改變 — 變數 [比較位置] 改到 下一個[比較位置] 要改變多少

↻

- 第1輪時，說出 4, 5 → 3, 4 → 2, 3 → 1, 2
- 第2輪時，說出 4, 5 → 3, 4 → 2, 3
- 第3輪時，說出 4, 5 → 3, 4
- 第4輪時，說出 4, 5

| 第幾輪 | 第1次 | 第2次 | 第3次 | 第4次 | 比較位置 |
|-----|-------------|-------------|-------------|-------------|---------|
| 1 | 4, <u>5</u> | 3, <u>4</u> | 2, <u>3</u> | 1, <u>2</u> | 5→4→3→2 |
| 2 | 4, <u>5</u> | 3, <u>4</u> | 2, <u>3</u> | | 5→4→3 |
| 3 | 4, <u>5</u> | 3, <u>4</u> | | | 5→4 |
| 4 | 4, <u>5</u> | | | | 5 |

氣泡排序(Bubble Sort)(康軒版)

2023/02/15 鹿鳴國中 黃天韻

| 說明 | 輸入 | 輸出 | 額外資料 |
|---------------------|-------------|-------------|------|
| 發完牌後，點擊「排序」鈕，進行氣泡排序 | 未排序的 紙牌清單[] | 已排序的 紙牌清單[] | |

解題策略 (BOTTOM UP 由下至上)

能交換紙牌清單[]中任 2 項的值 → 加入條件判斷 → 加入動畫 → 擴展程式至 3 張、4 張紙牌 → 尋找規律 → 重構程式 → 擴展程式至 n 張紙牌

設計 1：2 張牌，資料能夠互換，不考慮紙牌動畫

相關指令

| 指令 | Input | Output |
|----|-------|--------------------|
| | 編號 | 取得紙牌清單第「編號」的值(內容) |
| | 編號、新值 | 紙牌清單第「編號」的值(內容)=新值 |
| | 新值 | 變數「暫存」的值=新值 |
| | | 取得變數「暫存」的值 |

| Alg：交換 | Step | Process | Input | Output |
|--|------|-----------------------|-----------|-----------------------|
| <pre> graph TD A[暫存 = 清單[1]] --> B[清單[1] = 清單[2]] B --> C[清單[2] = 暫存] </pre> | 1 | 用「暫存」變數存放紙牌清單[1]的值 | 紙牌清單[1]的值 | 「暫存」變數的值=紙牌清單[1]的值 |
| | 2 | 設定紙牌清單[1]的值為紙牌清單[2]的值 | 紙牌清單[2]的值 | 紙牌清單[1]的值 = 紙牌清單[2]的值 |
| | 3 | 設定紙牌清單[2]的值為「暫存」變數的值 | 「暫存」變數的值 | 紙牌清單[2]的值 = 「暫存」變數的值 |
| | END | | | |

結果

| 程式 | 資料的變化 |
|----|-------|
| | |

設計 2：加入條件，根據條件決定是否交換
相關指令

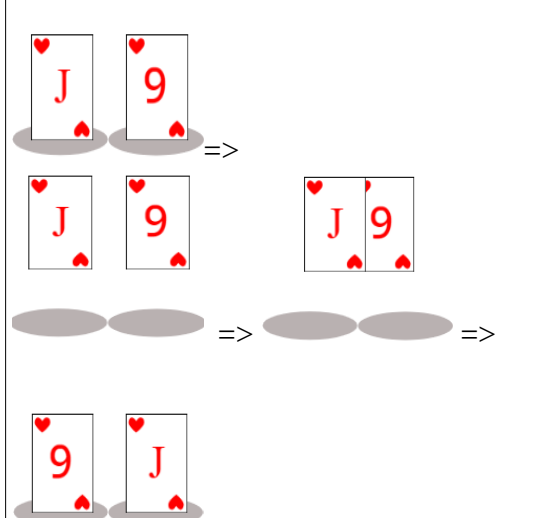
| 指令 | Input | Output |
|----|------------------|----------------------|
| | 編號 | 取得紙牌清單第「編號」的值(內容) |
| | 前數、後數 | 布林值(True 或 False) |
| | 條件、 成立時執行的指令塊 | 條件式值為「True」時執行包裹的指令塊 |

條件：

結果

| 程式 | 資料的變化 |
|----|-------|
| | |
| | |

設計 2.a 加入比較動畫、跟交換動畫，讓操作過程具體化，容易觀察、偵錯

| 程式 | 資料的變化 |
|--|--|
| <p>比較動畫呈現</p> <p>如果 紙牌清單 的第 2 項 < 紙牌清單 的第 1 項 那麼</p> <p>變數 暫存 設為 紙牌清單 的第 1 項</p> <p>替換 紙牌清單 的第 1 項為 紙牌清單 的第 2 項</p> <p>替換 紙牌清單 的第 2 項為 暫存</p> <p>交換動畫呈現</p> |  <p>The diagram illustrates the swap of two cards. It starts with two cards: Jack of hearts (J) and 9 of hearts (9). An arrow points to the next state where the cards are swapped: 9 of hearts (9) and Jack of hearts (J). A second arrow points to a final state where the cards are shown as a single combined unit, representing the result of the swap operation.</p> |

設計 3 延伸：利用目前程式完成 3 張牌的比較

| 程式 | 資料的變化 | | | | | | | | | | | | | | | | | | |
|--|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| <p>比較動畫呈現</p> <p>如果 紙牌清單 的第 3 項 < 紙牌清單 的第 2 項 那麼</p> <p>變數 暫存 設為 紙牌清單 的第 2 項</p> <p>替換 紙牌清單 的第 2 項為 紙牌清單 的第 3 項</p> <p>替換 紙牌清單 的第 3 項為 暫存</p> <p>交換動畫呈現</p> <p>比較動畫呈現</p> <p>如果 紙牌清單 的第 2 項 < 紙牌清單 的第 1 項 那麼</p> <p>變數 暫存 設為 紙牌清單 的第 1 項</p> <p>替換 紙牌清單 的第 1 項為 紙牌清單 的第 2 項</p> <p>替換 紙牌清單 的第 2 項為 暫存</p> <p>交換動畫呈現</p> <p>比較動畫呈現</p> <p>如果 紙牌清單 的第 3 項 < 紙牌清單 的第 2 項 那麼</p> <p>變數 暫存 設為 紙牌清單 的第 2 項</p> <p>替換 紙牌清單 的第 2 項為 紙牌清單 的第 3 項</p> <p>替換 紙牌清單 的第 3 項為 暫存</p> <p>交換動畫呈現</p> | <p>紙牌清單</p> <table border="1" data-bbox="893 963 1101 1120"> <tr><td>1</td><td>3</td></tr> <tr><td>2</td><td>1</td></tr> <tr><td>3</td><td>2</td></tr> </table> <p>=></p> <p>紙牌清單</p> <table border="1" data-bbox="893 1187 1101 1344"> <tr><td>1</td><td>1</td></tr> <tr><td>2</td><td>3</td></tr> <tr><td>3</td><td>2</td></tr> </table> <p>=></p> <p>紙牌清單</p> <table border="1" data-bbox="893 1411 1101 1568"> <tr><td>1</td><td>1</td></tr> <tr><td>2</td><td>2</td></tr> <tr><td>3</td><td>3</td></tr> </table> | 1 | 3 | 2 | 1 | 3 | 2 | 1 | 1 | 2 | 3 | 3 | 2 | 1 | 1 | 2 | 2 | 3 | 3 |
| 1 | 3 | | | | | | | | | | | | | | | | | | |
| 2 | 1 | | | | | | | | | | | | | | | | | | |
| 3 | 2 | | | | | | | | | | | | | | | | | | |
| 1 | 1 | | | | | | | | | | | | | | | | | | |
| 2 | 3 | | | | | | | | | | | | | | | | | | |
| 3 | 2 | | | | | | | | | | | | | | | | | | |
| 1 | 1 | | | | | | | | | | | | | | | | | | |
| 2 | 2 | | | | | | | | | | | | | | | | | | |
| 3 | 3 | | | | | | | | | | | | | | | | | | |

設計 4 重構 利用帶參函數重新組織程式，減少程式碼

相關指令

| 指令 | Input | Output |
|----|-----------------------|---------------|
| | 函數名、參數名 1、參數名 2、函數指令體 | |
| | 前數、後數 | 依據給定參數執行函數指令體 |

原程式：

定義成函數：

| 程式 | 資料的變化 | | | | | | | | | | | | |
|------|--|------|------|------|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| | <table border="1"> <thead> <tr> <th>紙牌清單</th> <th>紙牌清單</th> <th>紙牌清單</th> </tr> </thead> <tbody> <tr> <td>1 3</td> <td>1 1</td> <td>1 1</td> </tr> <tr> <td>2 1</td> <td>2 3</td> <td>2 2</td> </tr> <tr> <td>3 2</td> <td>3 2</td> <td>3 3</td> </tr> </tbody> </table> | 紙牌清單 | 紙牌清單 | 紙牌清單 | 1 3 | 1 1 | 1 1 | 2 1 | 2 3 | 2 2 | 3 2 | 3 2 | 3 3 |
| 紙牌清單 | 紙牌清單 | 紙牌清單 | | | | | | | | | | | |
| 1 3 | 1 1 | 1 1 | | | | | | | | | | | |
| 2 1 | 2 3 | 2 2 | | | | | | | | | | | |
| 3 2 | 3 2 | 3 3 | | | | | | | | | | | |

設計 5 再延伸：利用目前程式完成 4 張牌的比較

| 程式 | 資料的變化 |
|----|-------|
| | |

設計 6 重構 2 發現規律→數值抽象成變數→發現重複規律→用重複結構改寫

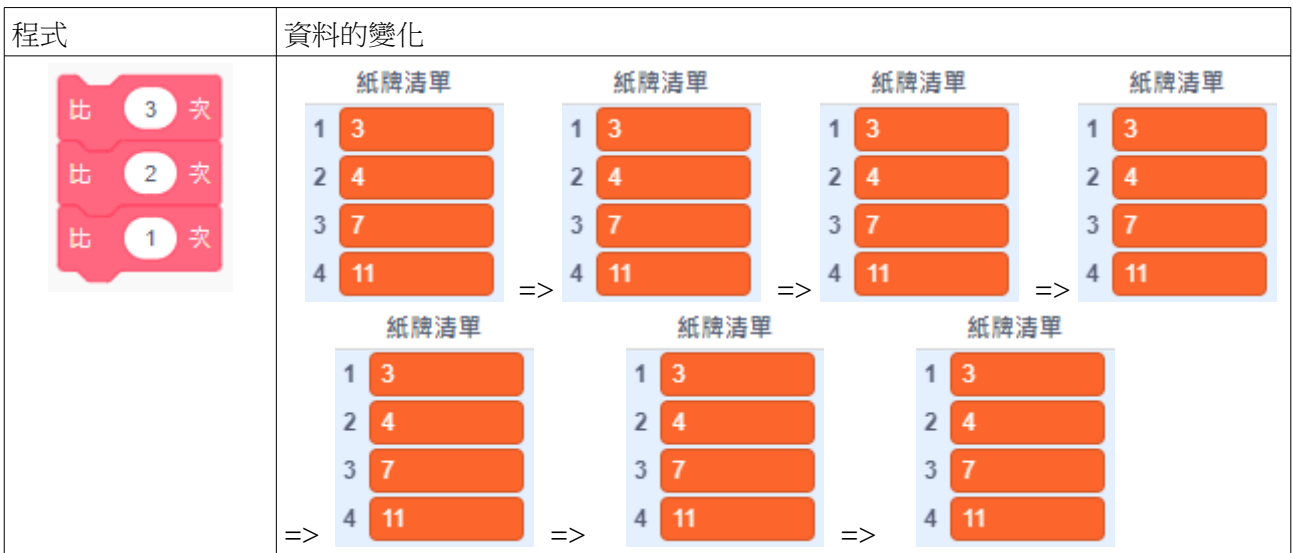
相關指令

| 指令 | Input | Output |
|----|-----------------|---------------------|
| | | 取得紙牌張數 |
| | 數值 | 設定變數「比較位置」的值為指定數值 |
| | | 取得變數「比較位置」的值 |
| | 變化值(正：遞增、負：遞減) | 「比較位置」=「比較位置」+ 變化值 |
| | 重複次數、 重複的指令體 | 包裹的指令塊重複執行直到指定次數才停止 |
| | 前數、後數 | 前數-後數 |

| 原程式 | 改迴圈後 |
|-----|------|
| | |



設計 7 將每 1 次的比較過程再封裝成帶參數函數



設計 8 重構 3 發現規律→數值抽象成變數→發現重複規律→用重複結構改寫

相關指令

| 指令 | Input | Output |
|----|-----------------|---------------------|
| | | 取得紙牌張數 |
| | 數值 | 設定變數「第幾輪」的值為指定數值 |
| | | 取得變數「第幾輪」的值 |
| | 變化值(正：遞增、負：遞減) | 「第幾輪」=「第幾輪」+ 變化值 |
| | 重複次數、 重複的指令體 | 包裹的指令塊重複執行直到指定次數才停止 |
| | 前數、後數 | 前數-後數 |

改寫

| 原程式 | 改迴圈後 |
|-----|------|
| | |

註：第幾輪是搭配範本動畫效果，初值從 1 起始，若初值為 3，就不必進行減法計算

設計 9 擴充至 n 張牌→數值抽象成變數

| 觀察數值 | 數值抽象成變數 |
|------|---------|
| | |