



逼真的照片

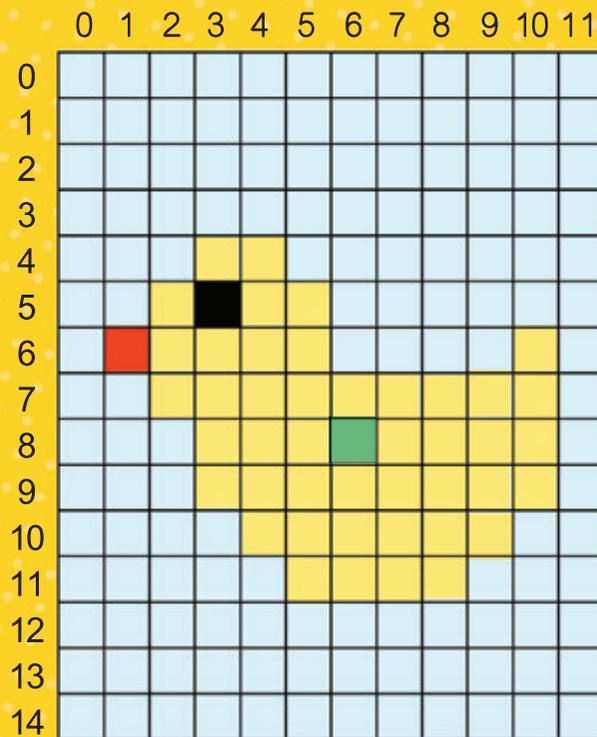
自從數碼相機面世之後，我們常在電腦上欣賞照片。可是你知道電腦是怎樣儲存圖像的嗎？程式又是怎樣讀取信息，然後把這些圖像在螢幕上顯示出來呢？

我們在電腦螢幕上看圖像的時候，所看到的，其實是感應器上分成數以百萬計的小正方形，稱為「像素」(pixel)。像素也是透過開關的形式來操作，它可以被打開，也可以被關閉。每一個單獨的像素可以為我們呈現出一種顏色，並有幾百萬種的顏色變化。

螢幕上，每一個像素都有各自特定的位置。比如右邊的這張圖像，網格左上角的那個像素的代號是(0,0)，而網格綠色的那個像素的代號是(6,8)。



那麼，鴨子的紅色嘴巴的代號就是(1,6)而它的黑色眼睛的代號是(3,5)。



電腦在儲存一個圖像檔案的同時，也存下了每一個像素的顏色和位置。當我們以圖像編輯軟件打開圖像檔案的時候，它就會讀取存在裏面的這些信息。

打開與關閉

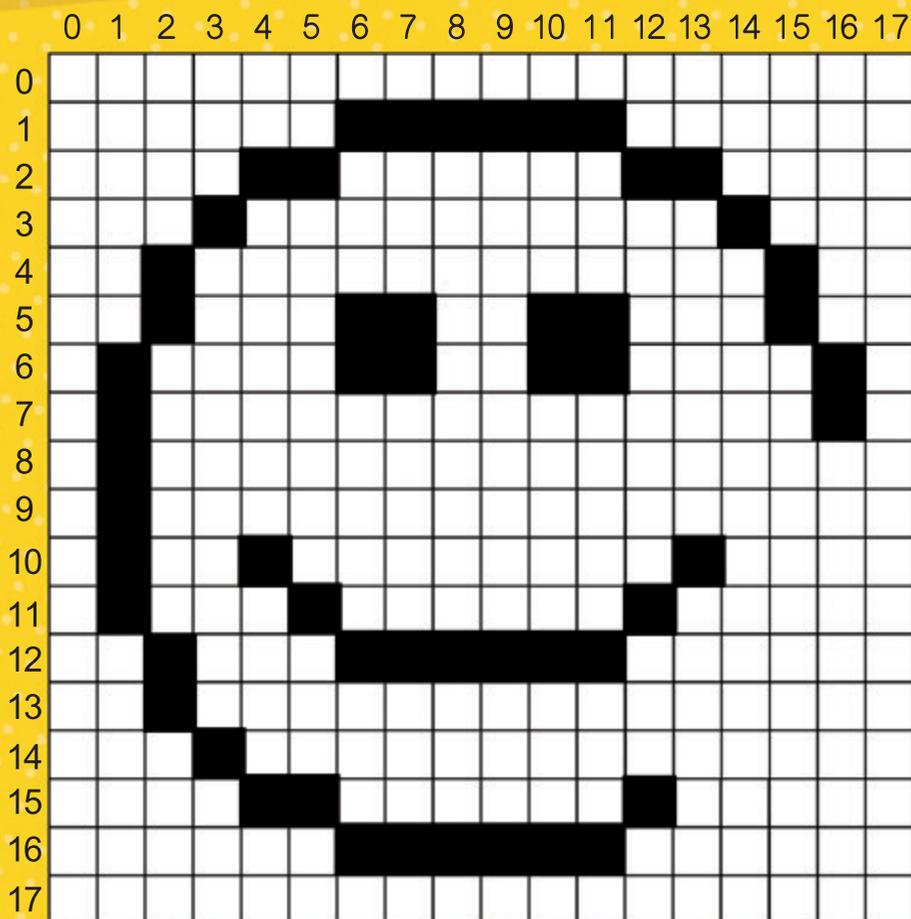
現在假設，你的電腦螢幕只能顯示黑與白這兩種顏色。它的像素可以打開，也可以關閉。打開的時候就是白色，關閉的時候就是黑色。

下面的這張圖是一個笑臉的圖像，可是淘氣的數據得來搗蛋，在圖像上把部份像素打開了！這令圖像變得不完整呢。小朋友，你認為哪些像素應該要關閉起來，才會令圖像完整呢？請你找出哪些像素需要被關閉，然後在適當的位置填上黑色吧。

(答案見第 98 頁)

數據得

記得當像素像關起來的時候，它就會顯示成黑色。





電影動畫

我們已經知道，電腦是怎樣顯示圖像的。那麼現在，請你仔細想想，我們可以用圖像來做些什麼呢？

原來，幾千張圖像可以組成一段動畫影片呢！當然，這些圖像得一張接着一張放映，播放的速度還得非常快喔！

我們在觀看影片的時候，每一秒會看到 30 張不同的圖像。因為圖像播放的速度很快，所以看起來就像是在移動一樣。

大家快來看看，

在 00:03:20 的時候，圖中的男孩在做什麼？



數據得

除了圖像，影片還可以帶有聲音呢！只要給電腦預先設定加入聲音檔的時間，它就可以在播放影片時，同時播放圖像和聲音了。

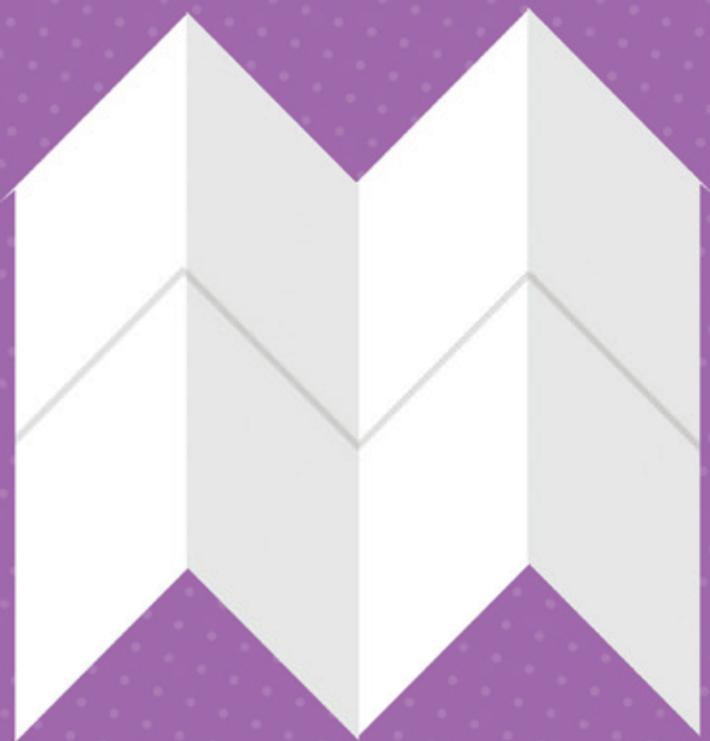


製作一本屬於你的動畫書吧！

你想體驗圖像如何化成動畫嗎？小朋友，我們可以一起試動手製作一本動畫書，這樣你就會更容易明白動畫的製作原理了。

材料：

- 1 張 A4 紙
- 鉛筆
- 剪刀
- 釘書機



步驟：

1. 將一張 A4 紙對摺一半，再對摺，然後重複對摺第三次。
2. 將紙打開，沿着摺痕將它剪成八張小紙張。
3. 在第 1 張紙張上畫一枚火箭；
4. 在第 2 張紙張上再畫一枚相同的火箭，不過它的位置則稍微往旁邊移動一些；
5. 在第 3 張到第 8 張紙張上重複畫出火箭，每畫一枚就往旁邊移動一些。
6. 按照 1 到 8 的順序將紙張整齊疊放，然後用釘書機固定在一側。
7. 最後，你可以快速翻動這 8 張紙張，這樣你的第一本動畫書已經製作完成啦！

小朋友，你能再多做一本動畫書，給我們講個故事嗎？



下達指令

在開始編寫一個程式之前，我們必須先思考需要電腦做什麼事情。接着，我們要將那件事情分拆成一系列指示。因為電腦無法自己思考，我們必須詳細地思考進行任務的事件，然後給電腦按順序列出每一個步驟（sequence）。

比如說，我們為一個機械人設計程式，讓它負責給小狗餵食，那麼我們就得考慮到以下這些問題，這樣我們才能知道，應該給機械人下達哪些指令：

- 小狗什麼時候會吃東西？
- 小狗會吃多少東西？
- 小狗吃什麼東西？
- 東西得裝在什麼容器裏？
- 小狗在哪裏吃東西？

當我們回答了以上所有的問題之後，才能開始順序給它下達指令，建立演算法。比如：

下午 3 點，把一勺狗糧倒在廚房地板上的塑膠碗裏。

如果我們遺漏了其中的任何一點，程式就會出錯，可憐的小狗也會因此挨餓。



機械人也是電腦的一種，我們必須清楚告訴機械人，要去哪個具體的地方，才能找到東西。指明地點的方法有很多種，「坐標」(coordinate)是其中之一的方法。

我們可以透過坐標來準確地指出位置，電腦就會知道它要找的東西究竟在哪裏。

坐標由兩個數字組成：x 值和 y 值。x 值從左到右橫穿地圖，y 值則是從下到上的。在標示坐標的時候，我們會先列出 x 值，然後才列出 y 值。

在地圖上給小狗定位！

在這張坐標地圖上，有一隻小狗、一個塑膠碗，一桶狗糧，還有一個機械人。你知道他們的正確位置嗎？

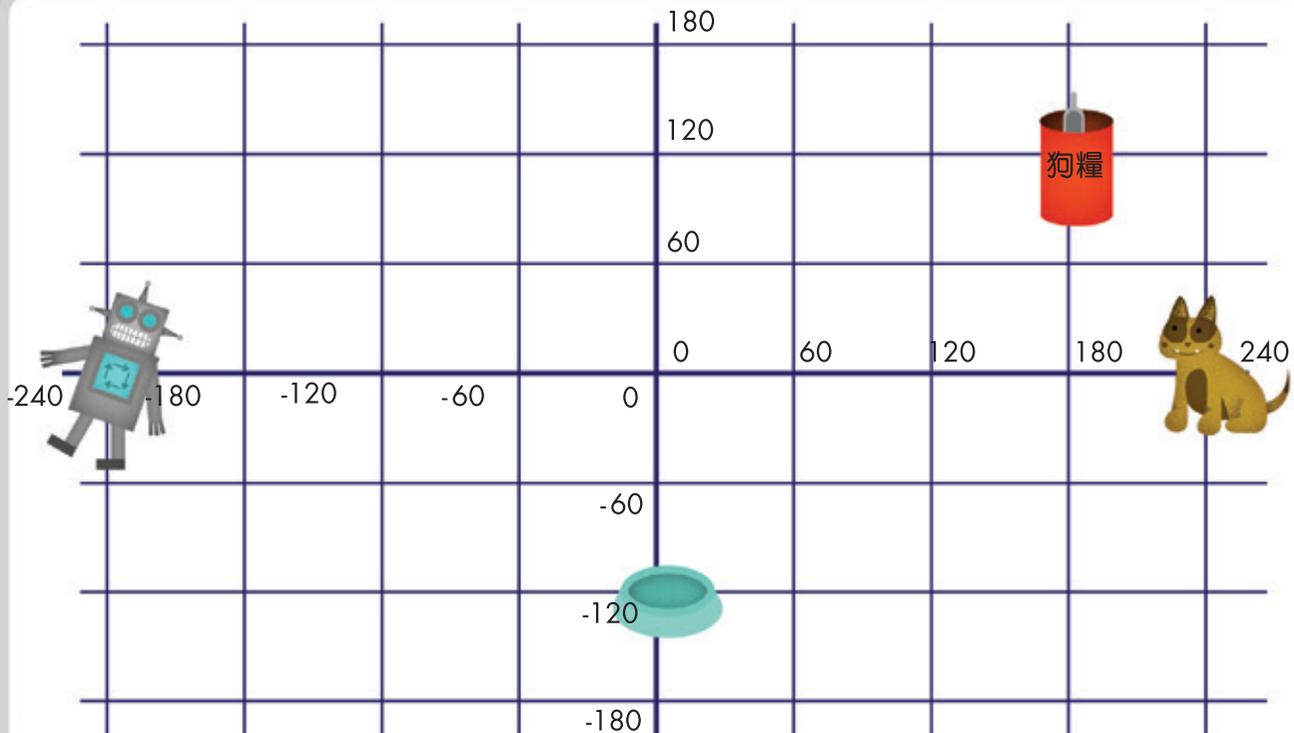
機械人所在位置的坐標是：
 $X=-240, y=0$

請在下面空白的位置寫出他們的坐標位置。

1. 狗糧：
2. 塑膠碗：
3. 小狗：

(答案見第 99 頁)

y 軸



x 軸



編寫程式

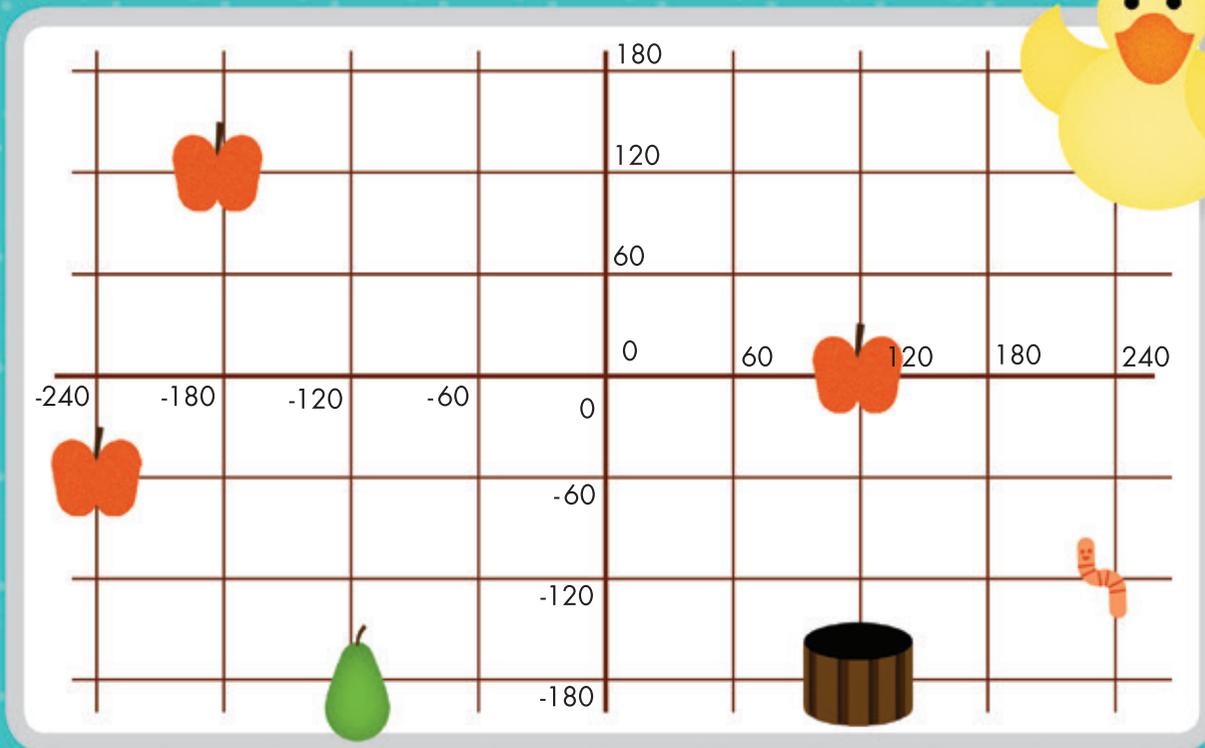
我們不僅學會了如何給電腦傳達詳細的指令，還學會了如何以坐標來給電腦提供準確的方位。所以現在，我們可以開始編寫程式啦！

摘蘋果

數據得正想去摘蘋果，他需要摘下 3 個蘋果來做午餐呢。小朋友，請你幫忙一起指揮他的行動吧。請從右面的指令方塊中選出正確的步驟，並按順序，在 內填上代表答案的英文字母。（答案可重複使用）

（答案見第 99 頁）

- A. 前往 $x=120,y=-180$
- B. 前往 $x=-180,y=120$
- C. 將籃子提起
- D. 前往 $x=-120,y=-60$
- E. 前往 $x=-240,y=-60$
- F. 前往 $x=120,y=0$
- G. 把蘋果放進籃子裏
- H. 前往 $x=60,y=-120$



y 軸

x 軸

正確的次序： → → → → → → →

有時候，電腦必須把同一個任務重複執行很多次。在這種情況下，程式員並不需要為每一次的任務單獨編寫指令，他只要在演算時使用迴圈（loop）——即「重複執行」指令，就可以做到了。

迴圈指令可以讓電腦不斷重複執行某個指令或序列。當然，程式員也可以給迴圈設定重複的次數。讓我來舉個例子吧！

假如現在，把你的朋友當成機械人，我們要他執行跳繩的指令。我們可以把這個指令設計成迴圈，讓「它」一直跳。

有了這個迴圈指令，「它」就會不停地跳啊跳，循環直到永遠，這可不行呢。所以，我們還得想辦法讓「它」停下，或者交給「它」另外一個任務。這時，我們就需要增加一個「條件」，當這個條件成立的時候，迴圈中的任務就會發生變化。

重複執行，直到永遠

跳繩

重複執行，直到感覺疲倦

跳繩

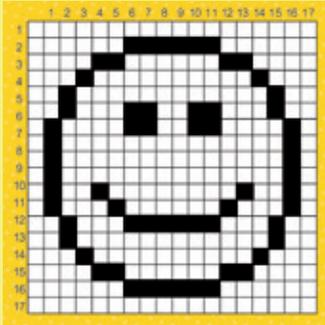
數據得

「條件」也同樣存在於我們的日常生活。比如說，做數學作業的時候，我們會把所有數值都加起來，直到沒有數值可以再相加。「沒有數值可以再相加」，就是一個條件。



答案

P.25



P.47

狗糧的坐標： $x=180, y=120$ 。

塑膠碗的坐標： $x=0, y=-120$ 。

小狗的坐標： $x=240, y=0$ 。

P.48

正確的次序： $A \rightarrow C \rightarrow F \rightarrow G \rightarrow E \rightarrow$

$G \rightarrow B \rightarrow G$