

桃園市 Scratch 程式設計推廣活動-程式說明文件

組別：■國中生活應用

題目:可視化智慧找零機器人

一、寫出這個任務要解決的問題或需求是什麼？

可視化智慧找零機器人可以讓書局可以推動「無人商店」，讓書局所需勞動力減少，並且增快結帳效率。

二、針對上述的要解決的問題或需求，你預計設計的角色 (Sprite)、變數、串列/清單 (list)、自訂積木 (函數)、廣播(程式流程)分別有哪些

角色：

中控、各項商品購買的按鈕、結帳按鈕、補充硬幣按鈕、長條圖繪製角色

變數：

系統-字串、系統-索引、系統-擷取數字(用來將用半形空白的字串拆分進清單或清單轉字串)

硬幣數量-1、硬幣數量-5、硬幣數量-10、硬幣數量-50(販售時顯示各面額硬幣數量)

硬幣總數(檢查總數是否超過80枚)

商品總價(紀錄客人購買的商品總價且可視化，不得超過300)

計算找零(計算之初設為商品總價，之後計算時會改變，直到計算完成)

找零硬幣數量(計算完找零後說出來的話)

銷售量前三(計算銷售量前三的商品分別是哪些)

A、B、C、D、E(從清單 銷售量中計算每個商品的銷售量，並繪製成長條圖)

縮放比例、最大長度、最大值(繪製長條圖所需變數)

i、j(索引)

清單：

暫存區(字串拆分後放置於此)

找零(計算所需找零的各種面額硬幣數量)

銷售量(每次購買時增加一個購買的商品至此清單)

銷售量-除錯(因為可以反覆購買，故銷售量不可重製，但是如果無法找足零錢，原本每次購買時所增加的銷售量還是有增加，因此需要銷售量-除錯來修正)

自訂積木 (函數)：

拆字串()、**清單轉字串**(用來將用半形空白的字串拆分進清單和清單轉字串)

計算銷售量(用來計算清單 銷售量中的各個商品數量，並加入進變數 A~E 中)

補充硬幣(用來補充硬幣)

銷售量-除錯(用來除錯的函式積木)

找最大值(用來找到銷售量前三和繪製長條圖)

畫長條圖、算出比例(用來繪製長條圖)

廣播(程式流程)：

重新、重新-補充硬幣判斷(用來交易完成後的重新交易)

結帳(按下結帳按鈕後廣播，讓中控計算找零)

補充硬幣(按下補充硬幣按鈕後廣播，讓中控詢問補充硬幣)

銷售量顯示(中控計算完找零後廣播，讓長條圖去繪製長條圖)

三、針對上述的要解決的問題或需求，分別以流程圖、虛擬碼或文字表示解決方法或過程。

重設

背景：

當按下綠旗時

重置變數、隱藏變數、重置清單、廣播訊息{重新}、廣播訊息{補充硬幣}

當收到訊息{重新}時

背景換成 初始 、隱藏變數、變數{商品總價}設為0

補充硬幣

中控：

當收到訊息{補充硬幣}

執行函式積木{補充硬幣}

函式積木{補充硬幣}：

隱藏變數{硬幣數量-1、硬幣數量-5、硬幣數量-10、硬幣數量-50、商品總價}

背景換成 文字

詢問硬幣現有數量

執行函式積木{拆字串(詢問的答案)}

判斷輸入數量是否為4(面額50、10、5、1)

判斷輸入硬幣總數是否超過80枚

函式積木{拆字串}：

將字串中各個數字分別添加進清單{暫存區}

購買商品

購買每個商品的按鈕：

當收到訊息{重新}

如果被按下

變數{商品總價}增加此商品價格

如果變數{商品總價}大於300

變數{商品總價}減少此商品價格

說出 總價不可超過300

否則(即變數{商品價格}小於等於300)

添加此商品(A~E)到清單{銷售量}

當收到訊息{結帳}

隱藏

停止這個物件的其他程式

結帳

結帳：

當收到訊息{重新}

隱藏

如果變數{商品總價}>0，那麼

顯示

如果被按下

廣播訊息{結帳}

隱藏

停止這個物件的其他程式

停止這個程式

中控：

當收到訊息{結帳}

The image displays four segments of Scratch code blocks, organized into two columns and two rows, detailing a coin change algorithm. The code is written in Chinese and uses standard Scratch block types: orange for loops and conditionals, green for variables, and blue for messages.

Top Left Segment:

- 刪除 找零 的所有項目
- 變數 計算找零 設為 無條件捨去 數值 商品總價 / 100 * 100 - 商品總價
- 如果 硬幣數量-50 > 計算找零 / 50 那麼
- 添加 無條件捨去 數值 計算找零 / 50 到 找零
- 變數 硬幣數量-50 改變 -1 * 無條件捨去 數值 計算找零 / 50
- 變數 計算找零 改變 -50 * 無條件捨去 數值 計算找零 / 50
- 否則
- 添加 硬幣數量-50 到 找零
- 變數 硬幣數量-50 改變 -1 * 硬幣數量-50
- 變數 計算找零 改變 -50 * 硬幣數量-50

Top Right Segment:

- 如果 硬幣數量-10 > 計算找零 / 10 那麼
- 添加 無條件捨去 數值 計算找零 / 10 到 找零
- 變數 硬幣數量-10 改變 -1 * 無條件捨去 數值 計算找零 / 10
- 變數 計算找零 改變 -10 * 無條件捨去 數值 計算找零 / 10
- 否則
- 添加 硬幣數量-10 到 找零
- 變數 硬幣數量-10 改變 -1 * 硬幣數量-10
- 變數 計算找零 改變 -10 * 硬幣數量-10

Bottom Left Segment:

- 如果 硬幣數量-5 > 計算找零 / 5 那麼
- 添加 無條件捨去 數值 計算找零 / 5 到 找零
- 變數 硬幣數量-5 改變 -1 * 無條件捨去 數值 計算找零 / 5
- 變數 計算找零 改變 -5 * 無條件捨去 數值 計算找零 / 5
- 否則
- 添加 硬幣數量-5 到 找零
- 變數 硬幣數量-5 改變 -1 * 硬幣數量-5
- 變數 計算找零 改變 -5 * 硬幣數量-5

Bottom Right Segment:

- 如果 硬幣數量-1 > 計算找零 那麼
- 添加 計算找零 到 找零
- 變數 硬幣數量-1 改變 -1 * 計算找零
- 變數 計算找零 改變 -1 * 計算找零
- 否則
- 如果 計算找零 = 0 不成立 那麼
- 說出 無法找足零錢，請換小額錢幣支付 持續 1 秒
- 銷售量-扣除
- 廣播訊息 重新
- 廣播訊息 補充硬幣
- 停止 這個程式



函式積木{計算銷售量}：

變數{i}設為0

重複(清單{銷售量}的長度)次

變數{i}改變1

如果清單{銷售量}的第 i 項=A

那麼變數{A}改變1

如果清單{銷售量}的第 i 項=B

那麼變數{B}改變1

以此類推至 E

繪製長條圖

長條圖：

分別等比例繪製五個商品的長條圖

找銷售量前三

長條圖：

找 A~E 中的最大值(此為函式積木{找最大值})

變數 i 設為0

變數 j 設為0

重複(變數{最大值}))次

如果(最大值-i)=A

變數{銷售量前三}設為字串組合{銷售量前三} 文具用品組

變數 j 改變1

如果(最大值- i)=B

變數{銷售量前三}設為字串組合{銷售量前三} 美術用品組

變數 j 改變1

以此類推至 E

如果 $j > 2$

那麼停止這個程式