

# Huffman Algorithm 壓縮程式

## 開發環境

---

Microsoft Windows 7 專業版(Professional)

Microsoft .NET Framework, Version 4.6.01055

Microsoft Visual Studio Community 2015, Version 14.0.25431.01 Update 3

## 使用語言

---

Microsoft Visual C#

## 簡介

---

Huffman Algorithm 是一種無失真資料壓縮編碼的演算法，其原理是將檔案中出現頻率較高的字元使用較短的編碼，而出現頻率較低的字元使用較長的編碼。利用此種動態長度編碼方式取代每個字元固定長度的編碼，來降低整體檔案平均的編碼長度達到無失真壓縮的目的。

要取得每個字元的 Huffman Code 必須先計算每個字元的出現頻率來建立 Huffman Tree，而建立 Huffman Tree 的過程中必須使用 Min Heap 的資料結構來進行排序與插入移除，如此令 Huffman Tree 的建構能在  $O(n \log n)$  的時間複雜度完成。

由於不同的檔案經過 Huffman Algorithm 會產出不同的 Huffman Code，因此壓縮檔內必須存放 Huffman Code 的解碼表才能正確的進行解壓縮，導致壓縮檔勢必因為存放解碼表而增加額外的容量。所以已經經過壓縮處理的檔案格式如 rar、jpg 等，再使用 Huffman Algorithm 壓縮時所產生的壓縮率非常有限，甚至因解碼表大小超過壓縮率節省的空間，造成壓縮後的大小比原始檔案來的大的現象。

為了減少解碼表所佔用的空間，本程式實作的編碼方式為 Canonical Huffman Code，此種特殊的編碼是透過對原本的 Huffman Code 進行些微調整就可得到，可以有效的減少存放解碼表的空間。

## 測試檔案

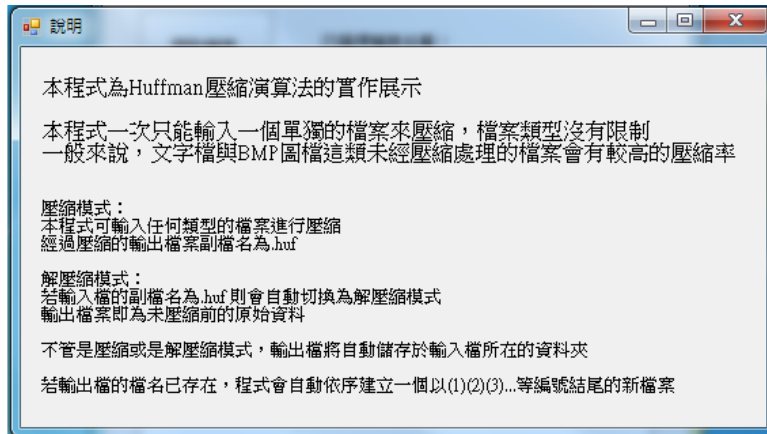
---

本作品額外提供三個輸入檔案作為壓縮測試檔之用：

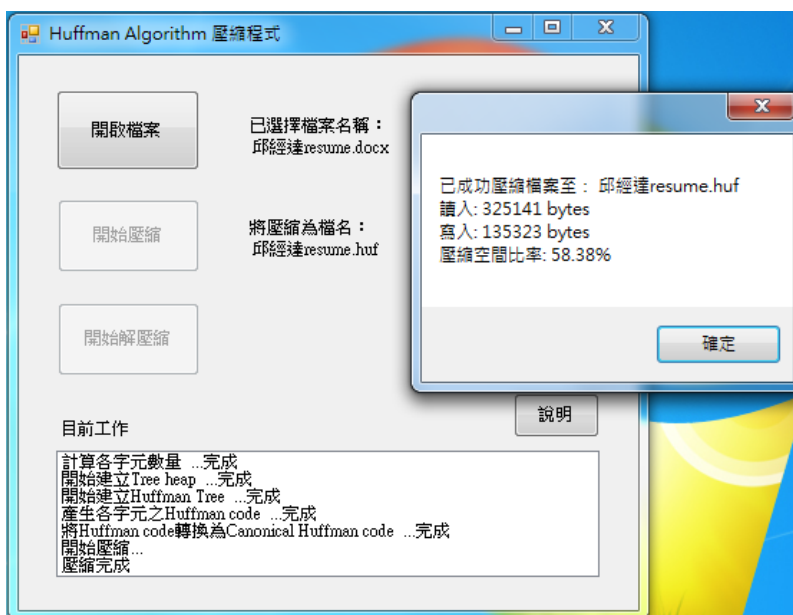
- 邱經達 resume.docx
- 邱經達 resume.pdf
- 邱經達 resume.bmp

## 程式介面與執行結果

### 說明



### 壓縮



### 解壓縮

