

2024 年【科學探究競賽-這樣教我就懂】

大專/社會組 科學文章表單

文章題目：霍夫曼編碼：數據壓縮中的利器

摘要：簡略介紹霍夫曼編碼，並利用實際例子來了解霍夫曼編碼是如何運作的。

文章內容：(限 500 字~1,500 字)

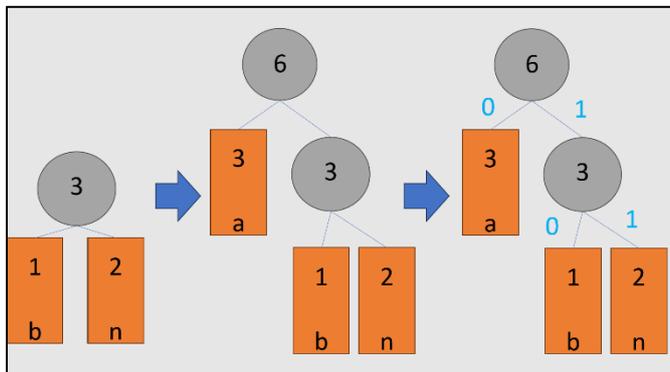
在這個網路的時代，資訊的傳遞與存儲已經成為我們日常生活中不可或缺的一部分。然而在面對這樣龐大的資訊量，資料的壓縮是不可避免的問題，你是否有想過我們的電腦是如何去進行壓縮的呢？

在進行資料壓縮最需要的兩點分別是：資料的完整性與資料的壓縮比。而「霍夫曼編碼(Huffman Coding)」就是一個不錯的方法。

霍夫曼編碼主要是利用字源的出現頻率來進行壓縮，出現越多次的字元則給予更短的二進制編碼，來讓整個檔案有更好的壓縮效果。這種編碼方法由大衛·霍夫曼於 1952 年提出，從此開啟了資訊壓縮領域的新篇章。

讓我們來看一個簡單的例子「banana」，在霍夫曼編碼中，我們首先要計算每種字元出現的次數，「a」出現了 3 次，「n」出現了 2 次，「b」出現了 1 次。因此我們給予「a」編碼 0，「n」編碼 11，「b」編碼 10，所以「banana」被編碼成了「100110110」，共有 9 位元。在這個例子中，如果將「a」給予了 2 位數，而「n」給予了 1 位數，整體編碼則會變成 10 為元，壓縮率會下降。

而決定每種字元編碼的方式，則是透過一種特殊的資料結構：霍夫曼樹(Huffman Tree)。這棵樹將會從葉節點開始生長，將出現頻率較低兩個的字元取出，建立一個新節點指向這兩個節點，並且將這個節點的出現頻率設為兩個子節點相加。重複執行直到剩餘一個節點。霍夫曼樹建完後，將其連接線分別給予 0 或 1，即可完成編碼。



圖表 1「banana」的編碼流程

霍夫曼編碼的優勢在於它能夠根據字元出現的頻率來動態地生成最佳編碼，從而實現了最優的壓縮效果。與固定長度編碼相比，霍夫曼編碼可以根據不同的資訊特性來靈活地調整編碼長度，從而更有效地壓縮資訊。如今，霍夫曼編碼已經運用在多個領域，像是檔案儲存

(zip)、圖像壓縮(jpeg)等都有利用到霍夫曼編碼。

然而霍夫曼編碼也有一些缺點，例如需要額外空間儲存該檔案對應的霍夫曼樹，如果此樹遺失，則會導致檔案無法被解碼。並且在壓縮、解壓過程中，需要花費額外的計算和存儲成本。因此實際應用中，多數情況會依不同情境利用不同的壓縮方式，來平衡壓縮成本與效果。

總的來說，霍夫曼編碼作為一種高效的資料壓縮方法，在當今資訊化社會中發揮著重要作用。因此，深入了解和應用霍夫曼編碼，將有助於我們更好地理解 and 利用資訊世界的密碼，從而推動資訊技術的不斷發展和創新。

參考資料

D.A. Huffman, "A method for the construction of minimum-redundancy codes", Proceedings of the I.R.E., sept 1952, pp 1098-1102.

https://web.archive.org/web/20050530145744/http://compression.graphicon.ru/download/articles/huff/huffman_1952_minimum-redundancy-codes.pdf

<https://hackmd.io/@Ben1102/SJGBYyo3O>

註：

1. 未使用本競賽官網提供「科學文章表單」格式投稿，**將不予審查**。
2. 字數沒按照本競賽官網規定之限 500 字~1,500 字，**將不予審查**。
PS.摘要、參考資料與圖表說明文字不計入。
3. 建議格式如下：
 - 中文字型：微軟正黑體；英文、阿拉伯數字字型：Times New Roman
 - 字體：12pt 為原則，若有需要，圖、表及附錄內的文字、數字得略小於 12pt，不得低於 10pt
 - 字體行距，以固定行高 20 點為原則
 - 表標題的排列方式為向表上方置中、對齊該表。圖標題的排列方式為向圖下方置中、對齊該圖