

The sequence of binary trees illustrates the construction of a Huffman tree from the initial nodes 4, 13, 12, 49, and 30. The trees are built iteratively by merging the two smallest nodes at each step. The final tree is a complete Huffman tree with a root node of 100.

The sequence of trees is as follows:

- Initial nodes: 4, 13, 12, 49, 30.
- Tree 1: Root 13 (left: 4, right: 9).
- Tree 2: Root 26 (left: 13, right: 12).
- Tree 3: Root 49 (left: 26, right: 12).
- Tree 4: Root 78 (left: 49, right: 30).
- Tree 5: Root 108 (left: 78, right: 30).
- Tree 6: Root 138 (left: 108, right: 12).
- Tree 7: Root 168 (left: 138, right: 30).
- Tree 8: Root 198 (left: 168, right: 30).
- Tree 9: Root 228 (left: 198, right: 30).
- Tree 10: Root 258 (left: 228, right: 30).
- Tree 11: Root 288 (left: 258, right: 30).
- Tree 12: Root 318 (left: 288, right: 30).
- Tree 13: Root 348 (left: 318, right: 30).
- Tree 14: Root 378 (left: 348, right: 30).
- Tree 15: Root 408 (left: 378, right: 30).
- Tree 16: Root 438 (left: 408, right: 30).
- Tree 17: Root 468 (left: 438, right: 30).
- Tree 18: Root 498 (left: 468, right: 30).
- Tree 19: Root 528 (left: 498, right: 30).
- Tree 20: Root 558 (left: 528, right: 30).

影片一：[https://video.search.yahoo.com/yhs/search;\\_ylt=AwrWn01deMRdsT0AYxIPxQt.;\\_ylu=X3oDMTByNWU4cGh1BGnVbG8DZ3ExBHBvcwMxBHZAaWQDBHNlYwNzYw--?p=Heapsort&fr=yhs-domaindev-st\\_emea&hspar=domaindev&hsimp=yhs-st\\_emea#id=2&vid=e2571267b1b1bf57b1a4c0509e056def&action=view](https://video.search.yahoo.com/yhs/search;_ylt=AwrWn01deMRdsT0AYxIPxQt.;_ylu=X3oDMTByNWU4cGh1BGnVbG8DZ3ExBHBvcwMxBHZAaWQDBHNlYwNzYw--?p=Heapsort&fr=yhs-domaindev-st_emea&hspar=domaindev&hsimp=yhs-st_emea#id=2&vid=e2571267b1b1bf57b1a4c0509e056def&action=view)

其中第一部影片主要介紹了HeapSort的邏輯思維過程，第二部影片是用C++語言打出的程式代碼。為了避免與網路內容過於相似，所以嘗試從C++的語法中思考將整個過程轉換為程式代碼的過程。

經過刪減不必要的變量，以及新增兩個while迴圈之後，整體結構與影片二的代碼靠攏

1. 在每一個def中都重複宣告了 `length = len(arr)` 導致一次次重新呼叫的時候 `length` 一直都沒有發生改變
2. 到現在還是不是很清楚原因的一點：我在使用if條件語句判斷的時候用到了and，我在這一步之前debug了很久很久，始終顯示 `out of range`，Google多次之後仍舊不知道怎麼更改，只好完全翻譯視頻中C++的語言，但是結果讓然沒變，還是 `out of range`。我把所有能和視頻一致的都該過來之後發現在互換and前後順序之後，程式不再提示有錯誤。但是仍舊不知道為什麼if條件語句判斷的and會有不同影響。之後會再進行研究。

3. 三個def之間的呼叫與return十分混亂，很難整理清楚三個def之間的呼叫關係以及每一個def所需要回傳的東西是什麼。