資料結構-作業3報告

* 使用環境:

使用Windows10作業系統，並且使用code-blocks編譯器撰寫程式碼

* 遇到問題與困境，與解決辦法:
* PROBLEM-1

1. 遇到困境(建立完huffman tree之後要用對應密碼對輸入字串加密):

密碼有長有短，因為huffman tree是strictly(full) binary tree不是perfect binary tree，所以由root出發給予路徑指向leaf的密碼大小不一，無法將每一個字元放入同樣大小的空間存取

1. 解決辦法:

原本打算使用array但是大小不一的情況下，放置進去若該項未存取則預設為0，這樣會影響最後輸出密文結果，後來打算動態宣告陣列但是compiler顯示conflicting types of function/initializer element is not constant，但是malloc語法與指標變數並無使用錯誤，所以果斷放棄改用queue資料結構來將各個字元code table放入，最後根據traverse輸入陣列後呼叫函數來將queue中，由於rear不斷根據輸入遞增，所以將queue中小於rear的index項輸出

* 程式內容:

1 先將建立huffman tree的結構跟linked-list指標還有max heap tree一一用函式寫出，建立節點與判斷大小存放機制

2 將需要判斷的數值加入生成huffman code的函數，並用先前寫的函數呼叫root到leaf的路徑用左0右1的方式加入arr陣列，這樣就完整建立huffman tree code table

3 把輸入陣列判定出現頻率並計算minimum weighted external path length

還有用上面提到的queue將根據index來把小於rear中存訪的陣列輸出

* 遇到問題與困境，與解決辦法:
* PROBLEM-2

1. 遇到困境(preorder和postorder輸出):
2. 解決辦法:

建立二元搜尋樹的過程相對第一題簡單，只需要將node判斷大小存放左右subtree，所以相對要面對的是建立樹之後的preorder跟postorder，就是用上課學到的方式分別是中左右/左右中，來把前序後序寫出

* 程式內容:

1 第一步當然是建立結構跟節點，還有一些inorder的函式根加入二元搜尋樹的操作，來讓比較數字大小及insert函數可以用linked-list表達出來並成功建立二元搜尋樹

2 再來寫出兩個function分別印出preorder / postorder，只是重複呼叫自身函數根據前序後序的規定次序來將左指標跟右指標依序指向