簡介：

Main()：

請使用者輸入指令，若指令為1，進行任務一建立最小堆積min heap；若指令為2，進行任務二建立最小最大堆積min-max heap；若都不是顯示"Command does not exit!"，並請使用者再輸入一次指令。

任務一：

呼叫read()進行讀檔，先讀完前三行標題後，利用tab個數的計算將資料擷取後依序放入vector型別的list中，並將每筆資料附上唯一序號，接者呼叫buildMinheap()，將list中的資料依據min heap的原則放入另一vector型別的heaplist中，方法是先將第0筆資料直接放入，之後的第i筆資料放入時先判斷i是否小於list的大小，若是則放入heaplist中，接者判斷第i筆的畢業生人數是否”小於”parent位置(第(i-1)/2筆)的畢業生人數，若是將第i筆及其parent位置交換資料，接者利用變數pos記錄parent位置，利用迴圈檢查第pos筆的畢業生人數是否”小於”pos其parent位置(第(pos-1)/2筆)的畢業生人數，直到沒有交換情形發生為止，將資料完全放入heaplist後，呼叫ans()分別將樹根(root)、底部節點(bottom)、最左下角落(leftmost)的序號、畢業生人數顯示於螢幕上。

任務二：

讀檔方式與任務一相同，接者呼叫buildMinMaxheap()，將list中的資料依據min-max heap的原則放入heaplist中，方法是先將第0筆資料直接放入，之後的第i筆資料放入時先判斷i是否小於list的大小，若是則放入heaplist中，呼叫isOdd()判斷第i筆資料屬於奇數層或偶數層，奇數層判斷第i筆的畢業生人數是否”大於”parent位置(第(i-1)/2筆)的畢業生人數，若是將第i筆及其parent位置交換資料，接者利用變數dad記錄parent位置，再利用變數dadG記錄dad的grandparent位置，利用迴圈檢查第dad筆的畢業生人數是否”大於”第dadG筆的畢業生人數，直到沒有交換情形發生為止，接者判斷第i筆的畢業生人數是否”小於”grandparent位置的畢業生人數，若是將第i筆及其grandparent位置交換資料，接者利用變數grandpa記錄grandparent位置，再利用變數grandpaG記錄grandpa的grandparent位置，利用迴圈檢查第grandpa筆的畢業生人數是否”小於”第grandpaG筆的畢業生人數，直到沒有交換情形發生為止；偶數層的判斷亦同，差別在於第i筆與parent的比較是”小於”，與grandparent的比較是”大於”，將資料完全放入heaplist後，呼叫ans()分別將樹根(root)、底部節點(bottom)、最左下角落(leftmost)的序號、畢業生人數顯示於螢幕上。

發現或心得：

到了上機練習時才發現輸出最左下角落(leftmost)的資料時有誤，原因是i的範圍介於0~size – 1，判斷式中少加了等號( i <= heaplist.size() – 1 )。