

Hw2 Report

Q1.

- 先設定INTERVAL = 12000, 以供random generate points和計算point是否在圓內使用
- 設定變數circle_points = 0, total_points = 0作為後續計算公式所需要的變數
- 根據題目提供的公式，撰寫child thread要執行的功能的function：
 - 使用rand()隨機產生 x, y ：
 - 將 (x, y) 固定在1*1的正方形中：
 - $x = (\text{rand}() \% (\text{INTERVAL} + 1)) / (\text{double})\text{INTERVAL};$
 - $y = (\text{rand}() \% (\text{INTERVAL} + 1)) / (\text{double})\text{INTERVAL};$
 - 計算 $(x^2 + y^2)$ 是否 > 1
 - < 1 代表 (x, y) 落在半徑= 1的圓內, 為circle point, 反之則為square point
 - 因為本題提供的公式為 $\pi = 4 * \text{circle points} / \text{total points}$, 所以只需計算circle point, total point的數量則是每次產生 $(x^2 + y^2)$ 都加一即可得到
 - 使用迴圈產生INTERVAL*INTERVAL個points以及更新完circle_points和total_points後, child thread的工作結束, 回到parent thread
- 回到parent thread後，根據 $\pi = 4 * \text{circle points} / \text{total points}$ 計算 π 並印出計算結果

Q2.

- 宣告global variable MAX和NUMBER_OF_THREAD紀錄要排序的數字個數以及threads的個數
- 宣告global array unsorted(紀錄尚未排序的數字)和sorted(用來記錄sort+merge後的結果)
- 根據題目要求，要製作兩個給thread使用的function：
 - sorting function
 - merging function
- sorting function要給兩個sorting thread使用，需要使用題目給的struct製作parameter紀錄sorting的起始位置跟結束位置，傳入sorting function後可以跟據起始位置跟結束位置將原本的array拆成兩半，前半和後半各自排序
- merge function則根據sort function排序好的兩半array進行merge
- 主程式中：
 - 使用rand()產生十個數字，store in unsorted
 - 宣告長度為3的thread array
 - 先create第一個sort thread
 - 再create第二個sort thread
 - 等待兩個thread都完成工作
 - 兩個sort threads都完成工作後，再create merge thread
 - 等待merge thread把其面兩個sort thread排序的結果進行合併，存入sorted
 - merge thread完成工作後，將結果印出來