

開發環境：

1. Windows10
2. IDE : CLion
3. Cygwin compiler (模擬 linux 環境)
p.s. `set(CMAKE_CXX_FLAGS_RELEASE "-O3")`
(compiler 設定有選 O3 的優化)
4. Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz 1.80 GHz

實作方法和流程&四種方法比較：

Task1：

bubble sort 每次先固定一個元素去跟其他元素比，若發現有值比它還要大的，就跟他做交換，直到它之後的所有元素都比過為止，就排序完成。

Task2：

首先，先切好使用者輸入的 k 份，得知每份大小，接著用 merge sort 的切割部分做切割，持續對半切，分成左右兩 list 以後持續對半切，直到切到一份大小為我先前 k 份每份要得到的那個大小就停(此部分用 thread 運作)，接著對每份數據做 bubble sort，並且用 merge sort 的 merge 部分開始做合併的動作。

Task3：

首先，先切好使用者輸入的 k 份，得知每份大小，接著用 merge sort 的切割部分做切割，持續對半切，分成左右兩 list 以後持續對半切，直到切到一份大小為我先前 k 份每份要得到的那個大小就停(此部分用 fork 運作，只對一邊做 fork，另一邊繼續跑)，接著對每份數據做 bubble sort，並且用 merge sort 的 merge 部分開始做合併的動作。

Task4：

首先，先切好使用者輸入的 k 份，得知每份大小，接著用 merge sort 的切割部分做切割，持續對半切，分成左右兩 list 以後持續對半切，直到切到一份大小為我先前 k 份每份要得到的那個大小就停，接著對每份數據做 bubble sort，並且用 merge sort 的 merge 部分開始做合併的動作。即 Task2 去除 thread 的部分。