

作業系統 HW2

B10605023 生機四 詹育晟

Q1

程式邏輯說明：

1. 引入必要的標頭檔與設定全域變數：
 - 使用 `pthread.h` 管理多執行緒。
 - 使用 `mutex` 保護全域變數 `points_in_circle`。
2. 主程式中執行流程如下：
 - 使用常數定義總點數 `TOTAL_POINTS` 及執行緒數 `NUM_THREADS`。
 - 透過 `pthread_create()` 建立多個子執行緒，平均分配亂數產生點的任務。
 - 每個執行緒負責產生一部分亂數點，並記錄落在圓內的數量。
 - 子執行緒執行完後透過 `pthread_join()` 等待所有執行緒完成。
3. 子執行緒執行的內容為：
 - 使用 `rand_r()` 產生亂數點 (x, y) ，範圍在 $[-1, 1]$ 。
 - 檢查點是否落在圓內。
 - 使用 `mutex` 更新全域變數 `points_in_circle`。
4. 最終由主執行緒輸出估算的 π 值：

執行指令說明：

1. 將程式儲存為 `hw2_Q1.c`
2. 編譯程式：`gcc hw2_Q1.c -o hw2_Q1`
3. 執行程式：`./Q1_out`

執行結果範例：

```
xavier1021@LAPTOP-68NFQUBM:/mnt/d/os_hw2/me$ gcc hw2_Q1.c -o Q1_out
xavier1021@LAPTOP-68NFQUBM:/mnt/d/os_hw2/me$ ./Q1_out
Estimated  $\pi$  = 3.141555
```

Q2

程式邏輯說明

1. 輸入處理：
 - 程式從命令列參數讀取一個整數 n ，代表欲產生的 Fibonacci 數列長度。
2. 記憶體配置：
 - 使用 `malloc()` 分配一段長度為 n 的整數陣列，指標命名為 `fib_sequence`，用來儲存產生的數列。
 - 因為這段記憶體位於 `heap` 區，主執行緒與子執行緒都可以共同讀寫。
3. 子執行緒行為：
 - 子執行緒由 `pthread_create()` 建立，進入 `generate_fibonacci()` 函式。
 - 在該函式中依序產生 Fibonacci 數列的每一項並儲存至 `fib_sequence[]` 中。
 - 完成後 `pthread_exit()` 結束該執行緒。
4. 主執行緒行為：
 - 呼叫 `pthread_join()` 等待子執行緒完成。
 - 印出 `fib_sequence` 的所有項目，並用逗號分隔輸出。

執行流程與指令：

1. 編譯程式：`gcc hw2_Q2.c -o Q2_out`
2. 執行範例（產生前 15 項）：`./Q2_out 15`

執行結果範例：

```
xavier1021@LAPTOP-68NFQUBM:/mnt/d/os_hw2/me$ gcc hw2_Q2.c -o Q2_out
xavier1021@LAPTOP-68NFQUBM:/mnt/d/os_hw2/me$ ./Q2_out 15
Fibonacci sequence:
0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377
```

Q3

程式邏輯與執行流程說明

1. 初始資料：

- 預設一組整數，例如：

```
int input_array[10] = {7, 12, 19, 3, 18, 4, 2, 6, 15, 8};
```

2. 結構體設計：

- 定義 ThreadData 結構體以傳遞陣列區段的 start 和 end 索引給各個排序執行緒：

```
typedef struct {  
    int start;  
    int end;  
} ThreadData;
```

3. 建立兩個排序執行緒：

- 對應到輸入陣列的前半段與後半段。
- 採用簡單的 Bubble Sort 完成區段內排序。

4. 合併執行緒：

- 使用雙指標（類似 merge sort 的合併步驟）將兩段排序後的資料依序合併成 sorted_array[]。

5. 主執行緒：

- 呼叫 pthread_create() 建立三個子執行緒。
- 使用 pthread_join() 等待三個子執行緒完成任務。
- 輸出最終排序完成的陣列。

執行流程與指令：

1. 編譯程式：cc hw2_Q3.c -o Q3_out
2. 執行範例（產生前 15 項）：./Q3_out

執行結果範例：

```
xavier1021@LAPTOP-68NFQUBM:/mnt/d/os_hw2/me$ gcc hw2_Q3.c -o Q3_out
xavier1021@LAPTOP-68NFQUBM:/mnt/d/os_hw2/me$ ./Q3_out
Sorted array:
2, 3, 4, 6, 7, 8, 12, 15, 18, 19
```