

Solving Minesweeper in parallel

第11組

311551131 賴俊宇 311551107 石偉辛 311552007 張千祐

1. Introduction/motivation

自電腦遊戲發展以來，踩地雷就是個經典的電腦遊戲，過往程式多是序列化執行求解地雷位置，因此我們想要試著用平行化的方法來處理這個問題，我們希望能夠透過多線程的方式做到平行化並達到更快的執行速度。

2. Statement of the problem

踩地雷，是一款經典的電腦遊戲。遊戲目標是找出所有沒有地雷的方格，完成遊戲。在數學上，踩地雷的複雜度為NP完全。

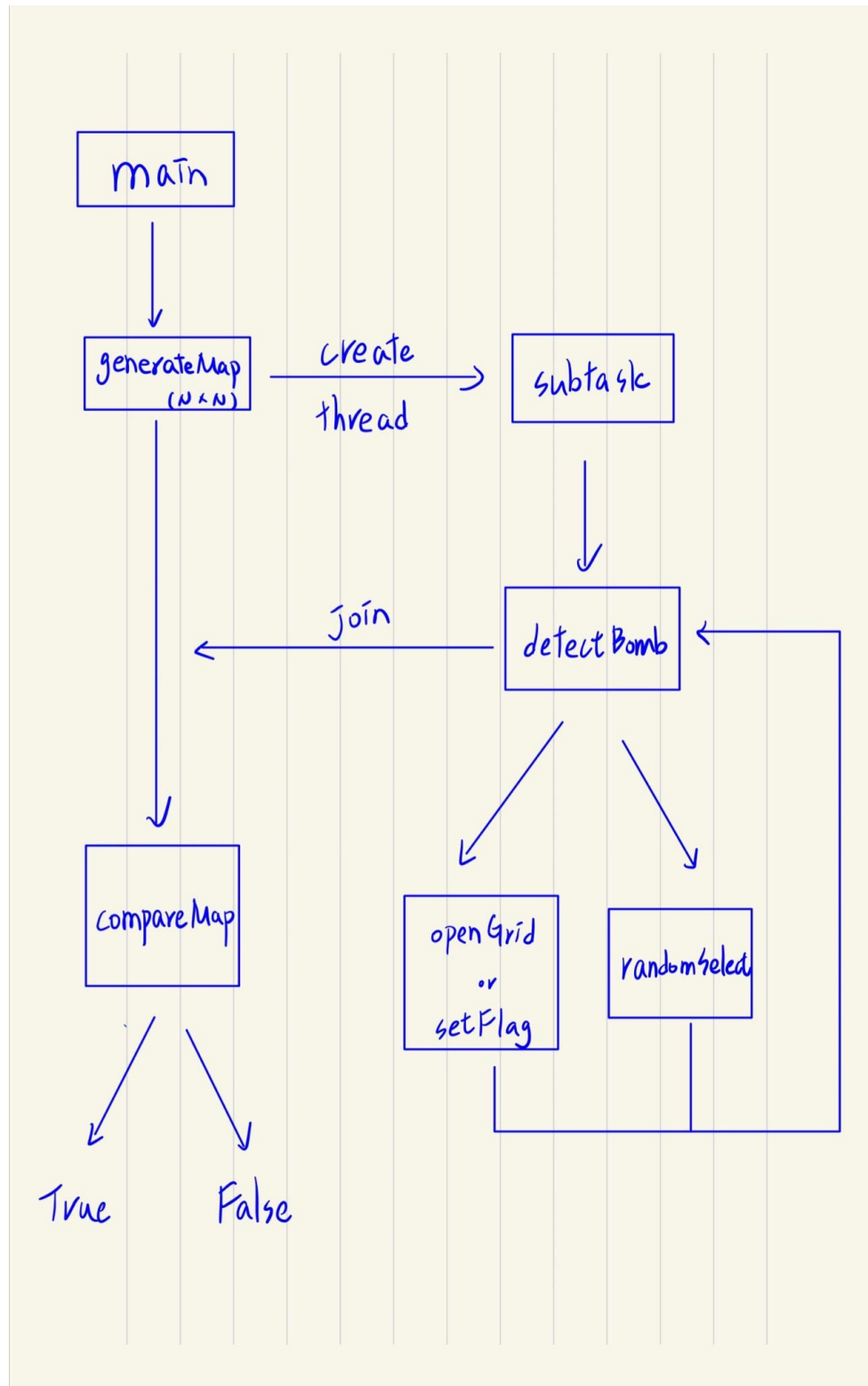
遊戲開始時，玩家可看到空白方塊排成的矩形，方塊數可由玩家自行選擇。如果是第一次點開方塊則不會踩到地雷。如果玩家點開沒有地雷的方塊，會有一個數字顯現其上，這個數字代表著鄰近八格的方塊有多少顆地雷(0~8)，玩家可在推測有地雷的方塊上，放置 flag 來標明地雷的位置，玩家須運用邏輯來推斷哪些方塊含或不含地雷。

我們希望設計一個程式，讓程式可以透過一些規則去判斷哪些空白格含有地雷或沒有地雷，並一步一步解開此遊戲。

3. Proposed approaches, which include the block diagram (or architecture) of your system. Please describe the function of each component, and the interaction among these components.

- A. `int **generateMap(N)` : 產生 $N \times N$ 大小的 truth map 及 map。
- B. `int main()` : 先呼叫 `generateMap` 之後，create 出子線程並等待所有子線程 join 之後呼叫 `compareMap`。
- C. `void *subtask(void *)` : 子線程函式，當中會呼叫 `randomSelect`, `detectBomb`, `openGrid`, `setFlag` 等函式。
- D. `void randomSelect()` : 隨機選擇 map 上的空白格點開，從 truth map 確認結果，並依序點開相鄰格子。
- E. `vector<pair<int, int>, int> detectBomb(int m, int n)` : 檢查 map 上的位置為 (m, n) 的數字格，判斷周圍空白格是否確定可解(確定可點開或確定是地雷)，將可點開或地雷格的位置資訊存入 vector 後回傳。
- F. `void openGrid(int m, int n)` : 將 (m, n) 的空白格點開，從 truth map 確認結果，更新到 map 中，並依序點開相鄰格子。

- G. void setFlag(int m, int n) : 在map當中將(m,n)這格設置flag, 意味著判斷這格為地雷, 並判斷目前map的地雷數量是否與 truth map 的地雷數量相同。
- H. bool compareMap() : 判斷 truth map 與 map bomb 位置是否皆相同。



4. Language selection

C++。

C++是一個我們較熟悉的环境，而且pthread也是一個我們習慣使用的套件。

5. Related work

- a. <https://www.geeksforgeeks.org/cpp-implementation-minesweeper-game/>
- b. <https://www.andrew.cmu.edu/user/cshan1/15418/>

6. Statement of expected results

- a. 建立完整踩地雷模型：能夠接收任意大小且有唯一解的踩地雷地圖，接著平行化計算找出所有地雷位置。
- b. 優秀的平行化：平行化後能確實加速程式，盡可能降低 dependency 產生的overhead，確保在增加thread數量的能夠同時維持盡量高的efficiency。

7. A timetable

- a. 11/10：完成 serial Minesweeper
- b. 12/1：完成 Minesweeper 平行化
- c. 12/9：presentation & report
- d. 1/3：完成書面 report

8. References

- a. [Creating a Minesweeper Map Generator in C++](#)
- b. <https://danq.me/minesweeper/>