Programming Assignment #3

0611031 資工 10 謝至恆

目標:

寫出一個踩地雷的程式，可以自己產生一個有地雷的棋盤，並有

Easy 9*9

Medium 16 * 16

Hard 30 * 16

3 種模式，

還需要寫一個由全部空白棋盤用 Propositional logic 演算得到解答的程式

一開始會先給予 round(sqrt(board_size)個已經打開的提示來讓程是有條件可以演算

解題思路:

需要一個 KB ，裡面包含 single-lateral clause 代表一開始安全的提示，並從 KB 中判斷每個格子中是安全或是有地雷。一值檢查剩餘的 clauses 如果 resolution 產生新的 clause 也要加進 KB 裡面

一個 KB0，一開始是空的，若 KB 判斷出格子是安全或有地雷，把這個位置和其 bool 值存到 KB0。

如果判斷出一個格子是 safe 則要跟生成遊戲的 module 要那一格的提示，

每個 cell 有兩個狀態(safe)沒有地雷，(mined)有地雷，用 bool 來表示

遊戲結束 ：

當所有格子都被打開或是被標記程地雷，且跟答案結果一致，代表成功

若最後剩下的格子有可能是地雷，則會無限跑一樣的迴圈，這時應該也算成功

其他表示失敗。

實作:

首先先生成實際的棋盤和玩家的棋盤，一個是知道所有安全位置和地雷(地雷是隨機的)，一個是全部未知的狀態，接著生成符合數目的提示，接下來就跟玩家的棋盤比較相關了，根據題是先產生相對應的 clause，之後先把周圍地雷是 0 的先打開，也就隨之產生新的 clause，這時候就要判斷有沒有可以 resolution 和 subsumpiton 的條件句，重複的 clause 和更嚴格的要把她刪除，才不會讓 clause 爆炸成長，一路到底就可以得到演算出來的棋盤，有時候會有解不出來的情況，也就是剩下的格子都有可能有地雷，這樣我門的演算法會一值跑相同的 clause 而且也沒辦法生成新的或消除 clause，發現這個狀況的時候就要將成是停止，

```
請輸入地雷模板要多大 : 9


Game Board :
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 0 0 0
1 * 1 2 * 3 1 0 0
1 1 1 2 * * 3 2 1
0 0 0 1 2 4 * * 1
1 1 1 0 0 2 * 3 1
1 * 1 0 0 2 2 3 1
1 1 1 0 0 1 * 2 *


hint :9
Play Board :
- - - - - - - - -
- - - - 0 0 - 0 0
- - - - 1 - - - -
- - - 2 - - - - -
- - - 2 - - - - -
- - 0 - - - - - -
- - - - - - - - -
- - - 0 - - - - -
- - - - - - - - -

遊戲開始（請輸入你要採的格子 格式（ 打開地雷 :o x y  標記地雷 :m x y ）

o 0 0
open
0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 0 0 0
- - - 2 - 3 1 0 0
1 1 1 2 - - 3 2 1
0 0 0 1 2 4 - - -
1 1 1 0 0 2 - - -
- - 1 0 0 2 - - -
- - 1 0 0 1 - - -
```

```
m 8 8
mark
8 8
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 0 0 0
1 ! 1 2 ! 3 1 0 0
1 1 1 2 ! ! 3 2 1
0 0 0 1 2 4 ! ! 1
1 1 1 0 0 2 ! 3 1
1 ! 1 0 0 2 2 3 1
1 1 1 0 0 1 ! 2 !
```

心得；

這次的作業我花了很多時間在看懂作業的解釋和要求，不知道是不是我的問題，一開始看的時候，不太懂是要可以自己操作還是要直接跑出演算法算出來的遊玩結果，後來大概看了 2~3 遍加上問同學才懂這次作業的架構，感覺範例的部分可以多給一點線索或是有個明確的表示方法，比較不會一開始做一頭霧水，最後又砍掉重練，這次的作業自己寫起來感覺有點殺雞焉用牛刀的感覺，感覺要算踩地雷不需要做的這麼複雜，不過主要目的應該是了解 propositional logic 的概念才對，這幾次作業下來我自己又多看了很多 c++的函數庫，和複習很多寫法，也算是蠻有收穫的。