

Sudoku Puzzle Homework

113321531 游志信

程式說明:

解題思路主要是利用Backtracking演算法, 將每種可能性塞入盤面, 只要算到不可行的結果就退回上個狀態, 程式透過stack來保存狀態(可以避免使用recursive所造成的stackoverflow), 嘗試將1~9填入盤面的空白格中, 直到解決所有空白格, 無法解決則無解

函數介紹:

print_board(board)這個函數會把數獨以可讀性高的方式印出來, 將3*3作為一個區塊切割, 邏輯很簡單

is_valid_input(board, row, col, num)這個函數會檢查數字能否放在指定的空格中, 按照數獨規則檢查, 直行1~9、橫列1~9、區塊1~9

```
# 檢查 3x3 方格的有效性
start_row = 3 * (row // 3)
start_col = 3 * (col // 3)
```

區分區塊的邏輯用的是//運算子, 其實就是floor除法, 假設row是2的話, 2//3會等於0(0.6666多), 乘上3是為了找到起始的那格

find_empty(board)這個函數會遍歷所有格子會傳遞第一個空格

solve_sudoku(board)這個函數使用backtracking演算法, 先初始化一個空的stack, 之後找到第一個空格並塞入stack, 接下來loop stack pop出來的空格, 嘗試塞入1~9的數字, 如果該數字可以放置, 則尋找下一個空格, 如果沒空格則解完數獨, 有的話保存當前狀態, 並記錄下個數字, 再將下一個空格的位置存入stack, 從1開始嘗試, 如果stack是空的但無法填滿盤面則無解

程式執行結果:

```
[Running] python -u "c:\Users\bear5\Desktop\sudoku_puzzle_113321531.py"
```

Original Board:

```
0 0 0 8 0 6 0 0 0
```

```
0 0 0 0 2 0 0 7 0
```

```
0 0 9 4 0 0 5 0 0
```

```
4 0 0 0 0 0 3 0 6
```

```
3 0 0 0 0 0 0 0 0
```

```
2 6 0 0 0 1 7 0 0
```

```
9 3 5 0 1 0 0 4 0
```

```
0 0 0 5 0 8 2 0 0
```

```
0 0 0 0 0 3 0 1 0
```

Solution:

```
7 4 3 8 5 6 1 9 2
```

```
5 8 6 1 2 9 4 7 3
```

```
1 2 9 4 3 7 5 6 8
```

```
4 9 1 7 8 5 3 2 6
```

```
3 5 7 2 6 4 9 8 1
```

```
2 6 8 3 9 1 7 5 4
```

```
9 3 5 6 1 2 8 4 7
```

```
6 1 4 5 7 8 2 3 9
```

```
8 7 2 9 4 3 6 1 5
```