```
1 using System;
 2 using System.Collections.Generic;
 3 using System.Linq;
 4 using System.Text;
 5 using System.Threading.Tasks;
 6 using System.Threading;
 7 namespace SudokuSolutionValidator
 8 {
 9
        class Program
10
11
            //27 Theards kiem tra Sudoku
            static Thread[] arrThread = new Thread[27];
12
13
            static bool valid = false;
14
            class paramaters
15
16
                public int row { get; set;}
17
                public int column { get; set; }
18
            //Ma tran Sudoku
19
20
            static int[,] sudoku =new int[9,9] {
21
                                             \{6, 2, 4, 5, 3, 9, 1, 8, 7\},\
22
                                             {5, 1, 9, 7, 2, 8, 6, 3, 4},
23
                                             \{8, 7, 7, 6, 1, 4, 2, 9, 5\},\
24
                                             \{1, 4, 3, 8, 6, 5, 7, 2, 9\},\
                                             \{9, 5, 8, 2, 4, 7, 3, 6, 1\},\
25
26
                                             \{7, 6, 2, 3, 9, 1, 4, 5, 8\},\
27
                                             \{3, 7, 1, 9, 5, 6, 8, 4, 2\},\
28
                                             {4, 9, 6, 1, 8, 2, 5, 7, 3},
                                             {2, 8, 5, 4, 7, 3, 9, 1, 6}
29
30
                                         };
31
32
            static void Main(string[] args)
33
            {
                DateTime startdate = DateTime.Now;
34
35
                int i, j,z, index=0;
36
37
                //9 Thread kiem tra dong
38
                for (i=0; i<9; i++)</pre>
39
40
                    arrThread[index] = new Thread(isRowValid);
41
                    arrThread[index++].Start(new paramaters() { row = i });
42
                    //isRowValid(new paramaters() { row = i });
43
                }
                //9 Thread kiem tra cot
44
45
                for (j = 0; j < 9; j++)
46
47
                    arrThread[index] = new Thread(isColumnValid);
                    arrThread[index++].Start(new paramaters() { column = j });
48
                    //isColumnValid(new paramaters() { column = j });
49
```

```
... udoku Solution Validator \backslash Sudoku Solution Validator \backslash Program.cs
```

```
2
```

```
50
51
                //9 Thread kiem tra 9 ma tran
52
                for (z = 0; z < 9; z++)
53
                {
54
                    int r = (z / 3) * 3, c = (z % 3) * 3;
                    arrThread[index] = new Thread(is3x3Valid);
55
56
                    arrThread[index++].Start(new paramaters() { row = r, column = →
57
                    //is3x3Valid(new paramaters() { row = r, column = c });
58
                }
                //Doi 27 ket thuc
59
60
                for (index = 0; index < 27; index++)</pre>
61
62
                    arrThread[index].Join();
63
                }
                if (valid)
64
65
                    Console.WriteLine("Sudoku Valid!!");
66
                else
                    Console.WriteLine("Sudoku Invalid!!");
67
68
                DateTime finishDate = DateTime.Now;
69
                TimeSpan time = finishDate - startdate;
                Console.WriteLine("Total:=" + time.Ticks);
70
71
                Console.ReadLine();
72
73
            static void isColumnValid(object param)
74
75
                paramaters p = (paramaters)param;
76
                for (int i=0; i<9;i++)</pre>
77
                {
                    for(int j=i+1; j<9;j++ )</pre>
78
79
                        if((1 > sudoku[i, p.column] || sudoku[i, p.column] > 9) || >
                        (sudoku[i,p.column]== sudoku[j, p.column]))
80
                            Console.WriteLine("Invalid in column: " + p.column);
81
82
                            valid = false;
83
                            return;
84
                        }
85
86
                }
87
            }
88
            //Phuong thuc kiem tra dong hop le
89
90
            static void isRowValid(object param)
91
92
                paramaters p = (paramaters)param;
93
                for (int i = 0; i < 9; i++)
94
                {
95
                    for (int j = i + 1; j < 9; j++)
                         if ((1 > sudoku[p.row, i] || sudoku[p.row, i] > 9)||
96
```

```
... udoku Solution Validator \backslash Sudoku Solution Validator \backslash Program.cs
```

```
3
```

```
(sudoku[p.row, i] == sudoku[p.row, j]))
 97
 98
                              Console.WriteLine("Invalid in row: " + p.row);
 99
                              valid = false;
100
                              return;
101
                         }
                 }
102
103
104
             //Phuong thuc kiem tra ma tran con hop le
             static void is3x3Valid(object param)
105
106
107
                 paramaters p = (paramaters)param;
                 int[] validityArray = new int[9] { 0, 0, 0, 0, 0, 0, 0, 0, 0 }; //
108
109
                 for (int i = p.row; i < p.row+3; i++)</pre>
110
                 {
111
                     for(int j = p.column; j< p.column+3; j++)</pre>
112
                     {
113
                         int num = sudoku[i, j];
                         if (num < 1 || num > 9 || validityArray[num - 1] == 1)
114
115
                              Console.WriteLine("Invalid in matix("+ p.row / 3+","+ →
116
                         p.column / 3+")");
117
                              valid = false;
118
                              return;
119
120
                         validityArray[num - 1] = 1;
121
                     }
122
                 }
123
             }
124
         }
125 }
126
```