22^{nd} Annual

Rowan University Programming Contest

hosted by the

Computer Science Department

Friday, 25 April 2008

22^{nd} Annual

Rowan University Programming Contest

hosted by the

Computer Science Department

Friday, 25 April 2008

Grid Placement Puzzles

Latin Squares

A ‘Latin Square’ is a square grid filled with symbols such that each symbol occurs exactly once in each row and column.

A	B	C	D	E
B	C	D	E	A
C	D	E	A	B
D	E	A	B	C
E	A	B	C	D

1	2	3	4
2	4	1	3
4	3	2	1
3	1	4	2

Sudoku

Sudoku adds a restriction to Latin squares: the grid is divided into subsquares, and each symbol can only appear once in each block, as well as each row and column.

B	E			C		I		A
	A				D			
D		G				B		H
		E	B					
				I	H	A		
	D				C			
			C	F			G	B
	G							C
I		C				F		D

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

Sudoku

Sudoku adds a restriction to Latin squares: the grid is divided into subsquares, and each symbol can only appear once in each block, as well as each row and column.

B	E	H	G	C	F	I	D	A
F	A	I	H	B	D	C	E	G
D	C	G	I	A	E	B	F	h
C	I	E	B	G	A	D	H	F
G	F	B	D	I	H	A	C	E
H	D	A	F	E	C	G	B	I
A	H	D	C	F	I	E	G	B
E	G	F	A	D	B	H	I	C
I	B	C	E	H	G	F	A	D

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

Sudoku-X

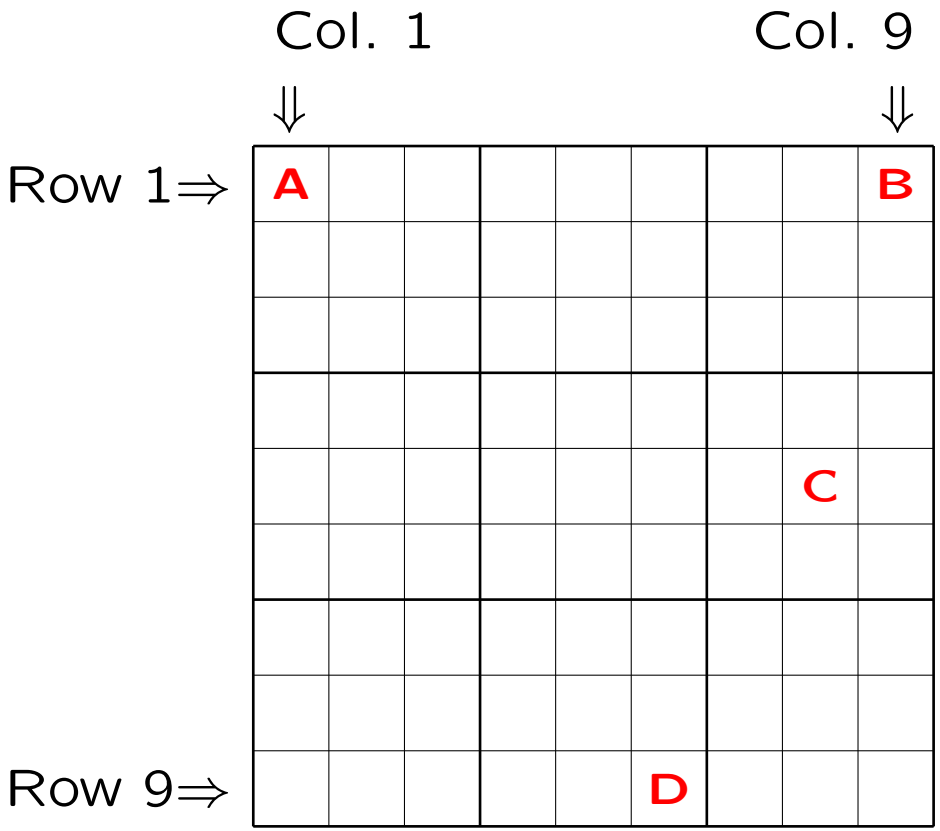
Sudoku-X adds another restriction to Sudoku: each symbol can only appear once on each major diagonal:

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

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

Some Terminology: Rows and Columns

Rows go across. Columns go up-and-down. Rows and columns will be numbered starting in the upper-left.

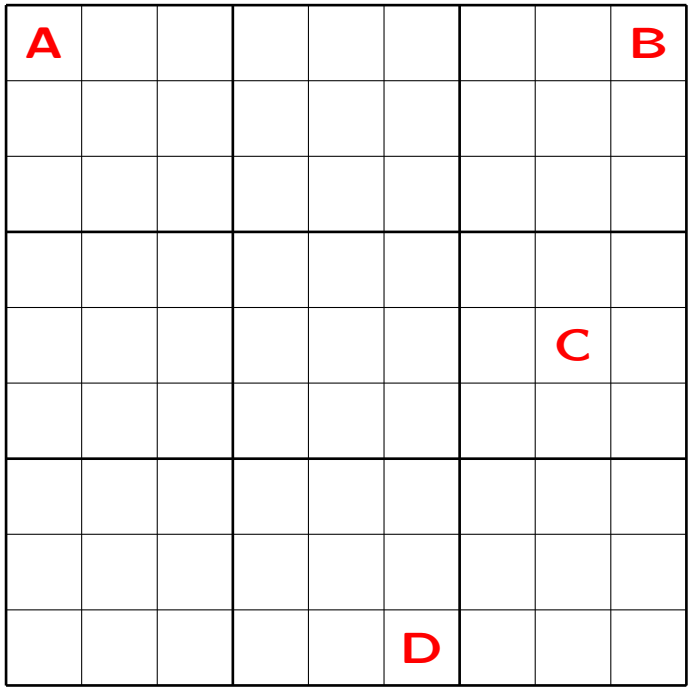


A is in Row 1, Column 1.
B is in Row 1, Column 9.
C is in Row 5, Column 8.
D is in Row 9, Column 6.

Some Terminology: Blocks

In Sudoku grids, Blocks will be numbered left-to-right, top-to-bottom:

For a 9×9 Sudoku:



Block Layout:

1	2	3
4	5	6
7	8	9

A is in Block 1.
B is in Block 3.
C is in Block 6.
D is in Block 8.

A 9×9 Sudoku has 9 Blocks; each Block has 9 spaces.

Some Terminology: Blocks

In Sudoku grids, Blocks will be numbered left-to-right, top-to-bottom:

For a 4×4 Sudoku:

A			
			B
		C	

Block Layout:

1	2
3	4

A is in Block 1.
B is in Block 2.
C is in Block 4.

A 4×4 Sudoku has 4 Blocks; each Block has 4 spaces.

Some Terminology

All square grids have two diagonals.
Diagonal 1 runs from upper-left to lower-right.
Diagonal 2 runs from lower-left to upper-right.

1								2
	1						2	
		1				2		
			1		2			
				X				
			2		1			
		2				1		
	2						1	
2								1

The **X** is in both Diagonal 1 and Diagonal 2.

Some Considerations

Every correct Sudoku-X solution is a correct Sudoku solution, and is also a correct Latin square.

Every correct Sudoku solution is also a correct Latin square, but may not be a correct Sudoku-X.

Not every correct Latin square is a correct Sudoku or Sudoku-X.

Sudoku squares must have a number of rows/columns which is a perfect square. (Remember that each block is itself a square.) Latin squares have no such restriction on the number of rows and columns.

Catching Errors: Duplicates in Columns

Sometimes, it's easy to tell if a mistake has been made:

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

Here, the solver has placed a second 5 in Column 3.

This kind of mistake is easy to catch.

Your program will have to catch errors of this sort.

Catching Errors: Duplicates in Rows

Sometimes, it's easy to tell if a mistake has been made:

3			
	2		1
4			**4**
	1	3	

Here, the solver has placed a second 4 in Row 3.

This kind of mistake is easy to catch.

Your program will have to catch errors of this sort.

Catching Errors: Duplicates in Blocks

Sometimes, it's easy to tell if a mistake has been made:

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

Here, the solver has placed a second 8 in Block 3.

This kind of mistake is easy to catch.

Your program will have to catch errors of this sort.

Catching Errors: Duplicates in Diagonals

Sometimes, it's easy to tell if a mistake has been made:

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

Here, the solver has placed a second 9 in Diagonal 1.

This kind of mistake is easy to catch.

Your program will have to catch errors of this sort.

Catching Errors: Impossible Configurations

Sometimes, it's not so easy to tell if a mistake has been made:

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

The solver has placed a **4** at (4,8), in Block 6.

Now there is no place for a **4** in Block 3.

Your program will **NOT** have to catch errors of this sort.

The Challenge

Write a program that tells whether a grid of numbers is:

Solved Sudoku-X completely filled in per Sudoku-X rules.

Unsolved Sudoku-X some blanks, but no improper duplicates.

Incorrect Sudoku-X incorrect duplicates were found.

Solved Sudoku completely filled in per Sudoku rules.

Unsolved Sudoku some blanks, but no improper duplicates.

Incorrect Sudoku incorrect duplicates were found.

Solved Latin Square completely filled in per Latin square rules.

Unsolved Latin Square some blanks, but no duplicates.

Incorrect Latin Square incorrect duplicates were found.

The Challenge

In the event improper duplicates are found, **all** of them **must** be listed.

If a grid fails a particular designation, it should still be checked against the less-restrictive designations. An incorrect Sudoku-X may still be a correct (or unsolved) Sudoku. An incorrect Sudoku may still be a correct (or unsolved) Latin square.

A grid that passes a particular designation does not need to be reported on again, because it will obviously pass the less-restrictive classifications.

Note: a grid whose number of rows/columns is not a perfect square should be treated only as a Latin square, since it cannot be a Sudoku at all.

The Challenge – an Example

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

```
Size: 9 x 9
Incorrect Sudoku-X
  2 is repeated on diagonal 1
    (1,1) (4,4) (8,8)
  9 is repeated on diagonal 2
    (5,5) (9,1)
  2 is duplicated in block 1
    (1,1) (2,3)
Incorrect Sudoku
  2 is duplicated in block 1
    (1,1) (2,3)
Unsolved Latin square
```