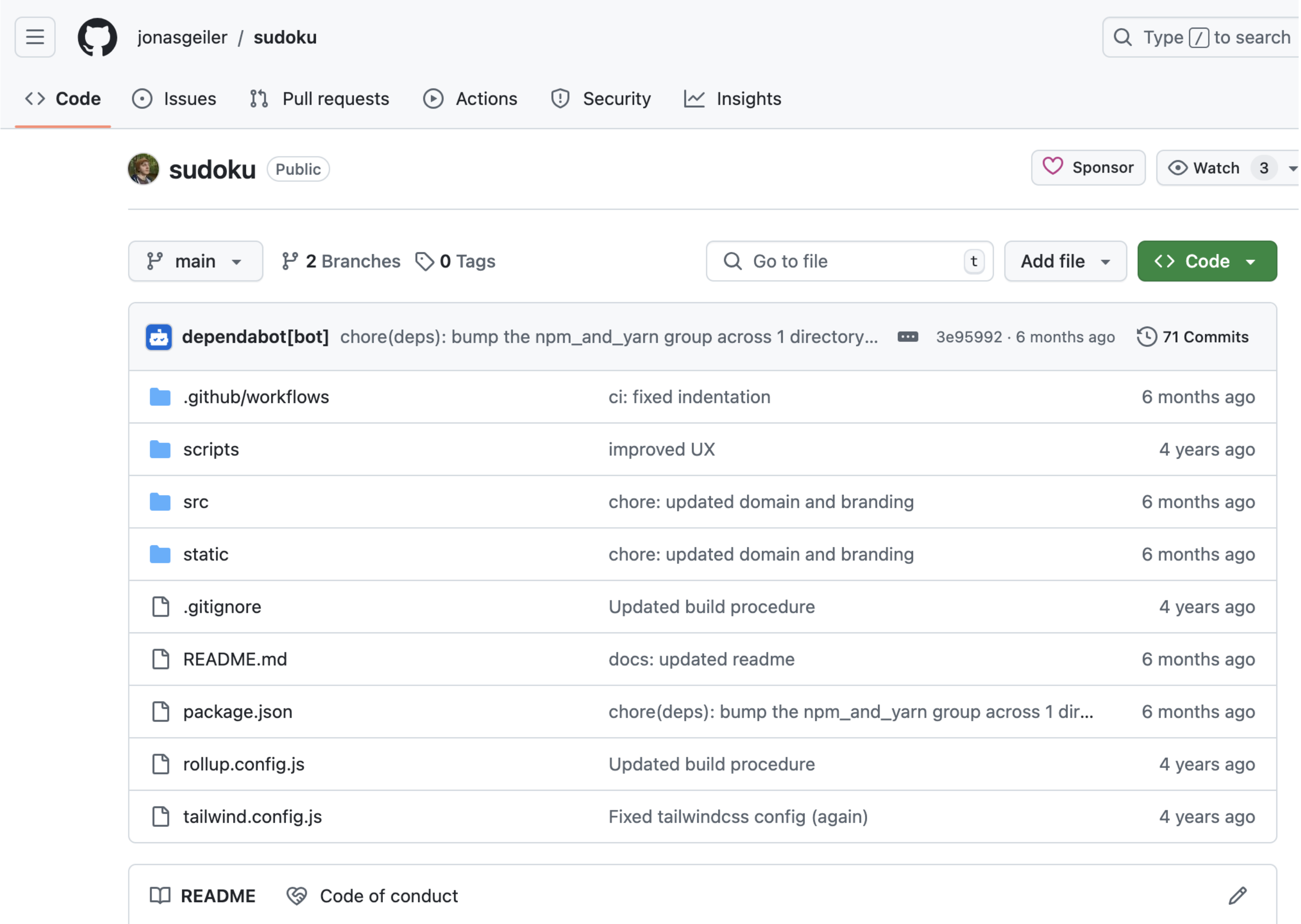
数独乐乐 OOAD

逆向分析现有数独项目的面向对象技术，然后按新业务愿景，改进升级为数独乐乐应用；分析、设计出面向对象技术方案，并落地代码实现。

现有数独项目：https://github.com/jonasgeiler/sudoku



dependabot[bot] chore(deps): bump the npm_and_yarn group across 1 directory... 3e95992 · 6 months ago 71 Commits

.github/workflows	ci: fixed indentation	6 months ago
scripts	improved UX	4 years ago
src	chore: updated domain and branding	6 months ago
static	chore: updated domain and branding	6 months ago
.gitignore	Updated build procedure	4 years ago
README.md	docs: updated readme	6 months ago
package.json	chore(deps): bump the npm_and_yarn group across 1 dir...	6 months ago
rollup.config.js	Updated build procedure	4 years ago
tailwind.config.js	Fixed tailwindcss config (again)	4 years ago

逆向分析

逆向工程，对现有项目的OOA、OOD、OOP进行分析：

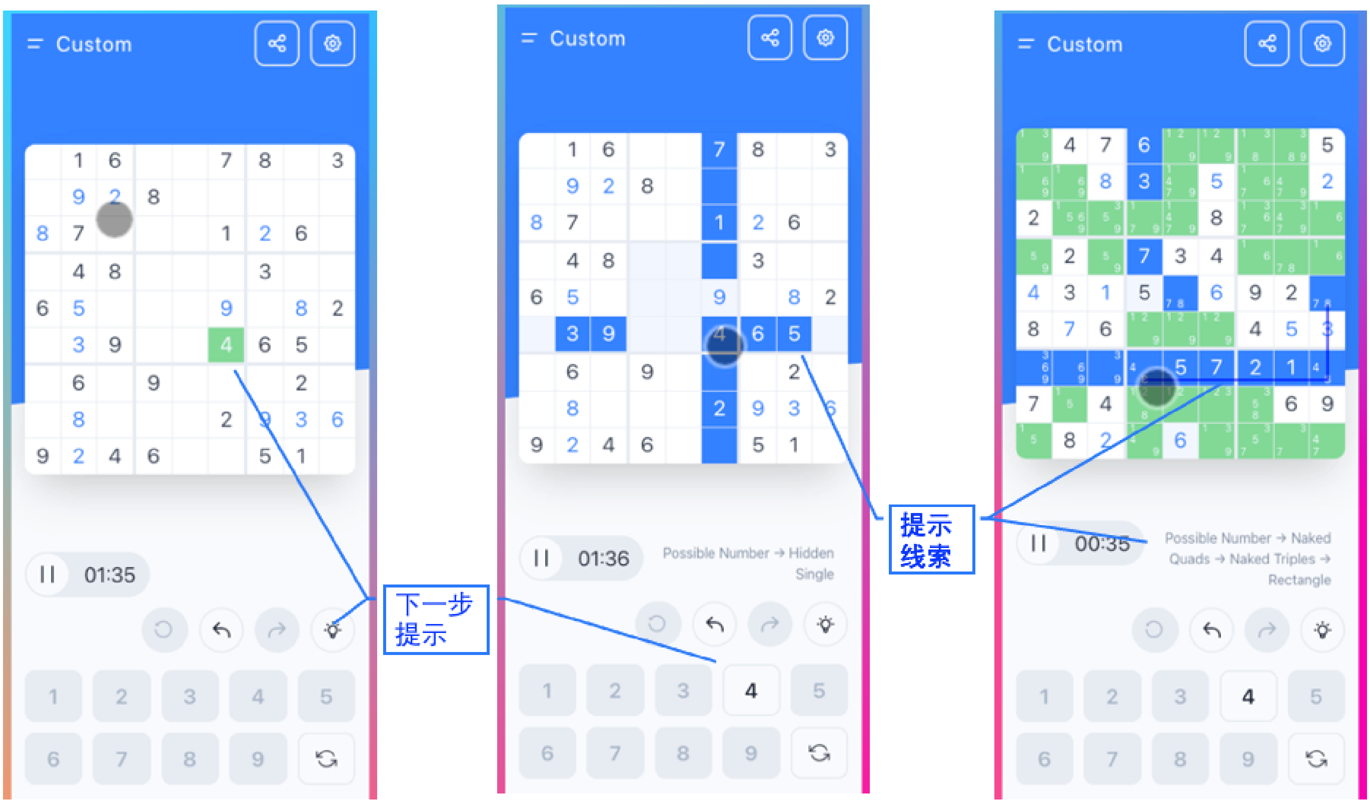
1. 讨论其设计思想、设计原则和使用的设计模式，给出其**愿景**、**用例分析**、**领域模型**、**技术架构与对象模型**；
2. 结合课程，评价现有OOD架构与设计的优劣，给出**改进建议。**

新业务愿景

升级应用为**数独乐乐**。让游戏更加适合初学者，给用户更多友好指引。同时，集成SudokuWiki.org，便于持续提升求解算法和验证开发。

下一步提示

通过提示按钮，为用户提示下一步可以选择的答案，并提示线索，说明推理答案所用的方法（策略）。



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

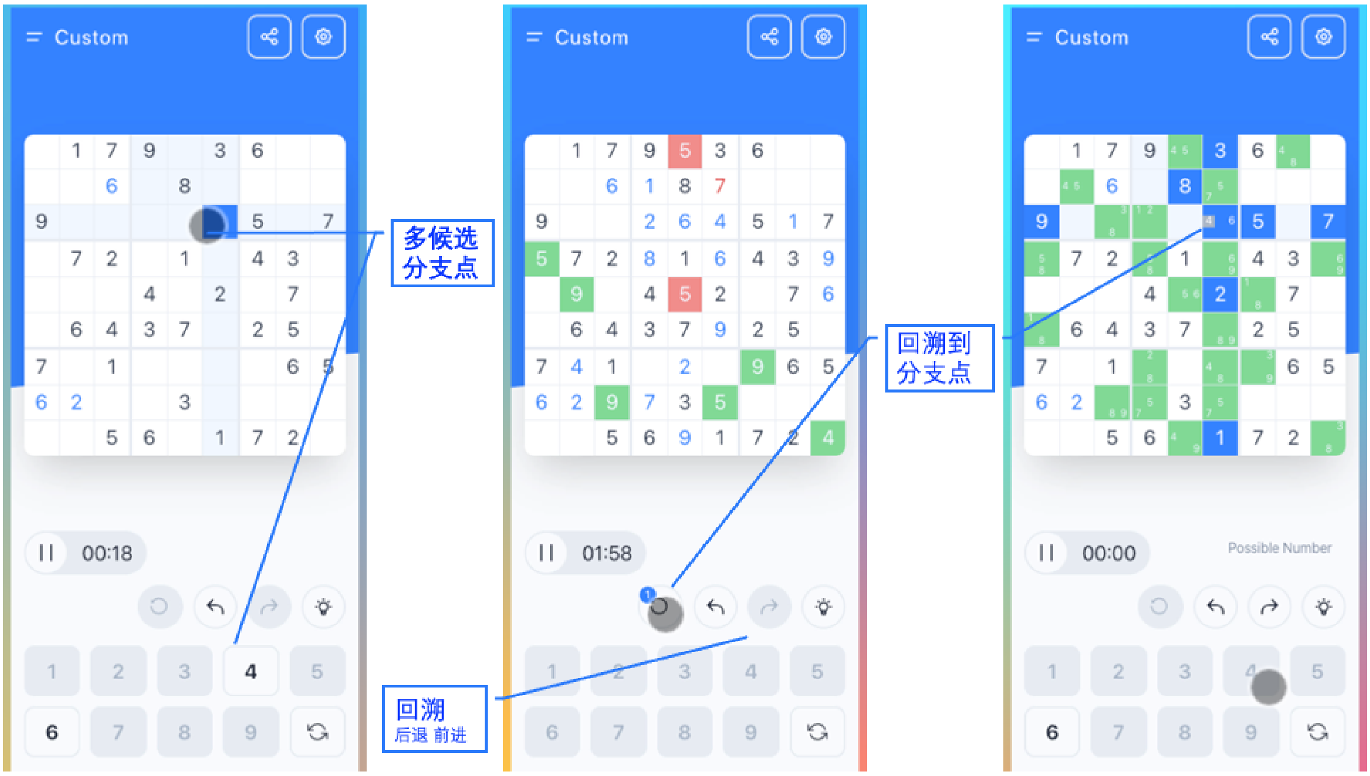
	1	6			7	8		3
	9	2	8					
8	7				1	2	6	
	4	8				3		
6	5				9		8	2
	3	9			4	6	5	
	6		9				2	
	8				2	9	3	6
9	2	4	6			5	1	

Possible Number → Hidden Single

Possible Number → Naked Quads → Naked Triples → Rectangle

探索回溯

方便用户在多种可能得答案中，漫游、探索、回溯

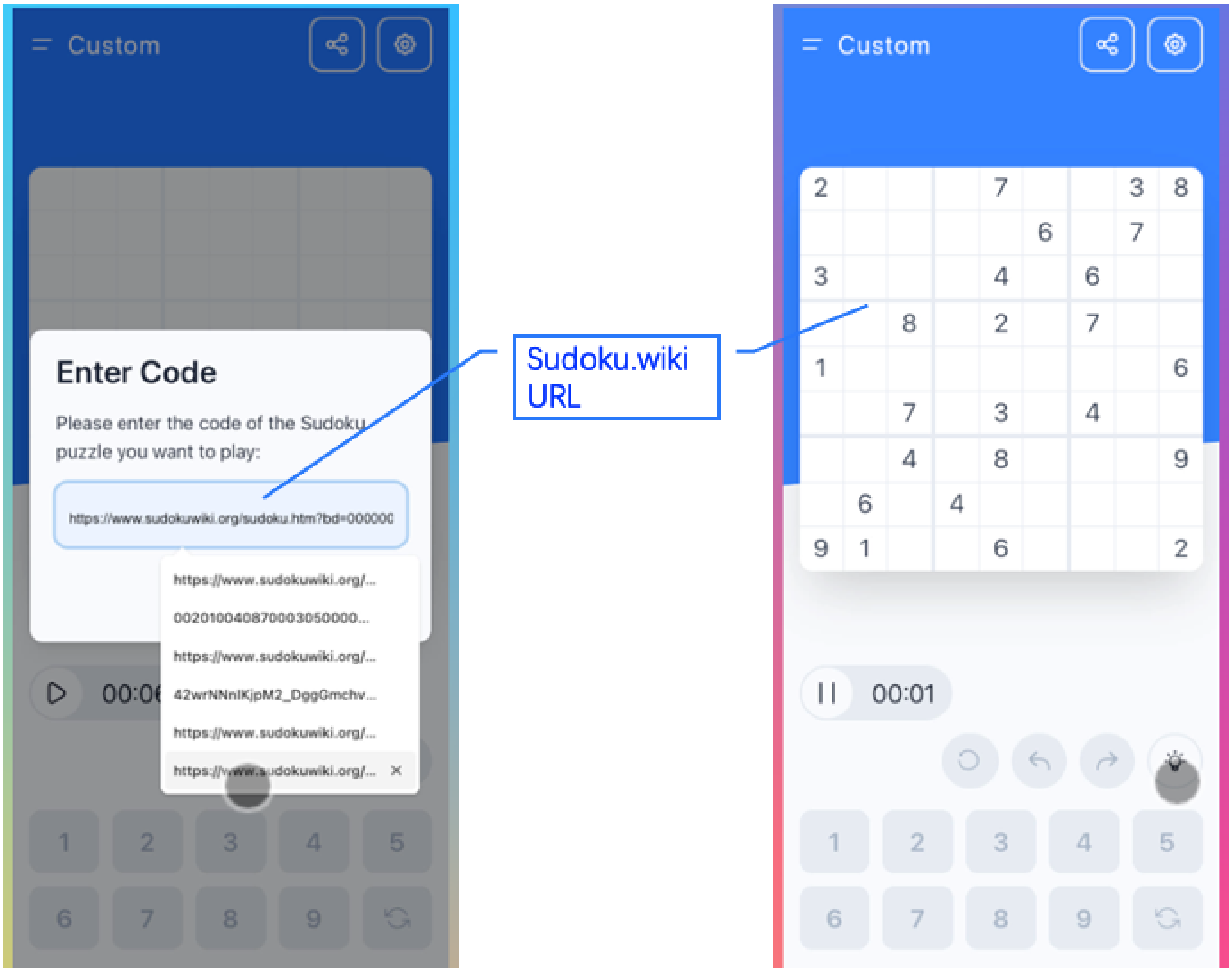


资源集成

集成数独Wiki资源，包括题目、算法策略。 https://www.sudokuwiki.org/Sudoku.htm

题目导入

能够将数独Wiki的题目页的URL作为Code，直接导入到数独乐乐中使用。



Enter Code

Please enter the code of the Sudoku puzzle you want to play:

https://www.sudokuwiki.org/sudoku.htm?bd=000000

https://www.sudokuwiki.org/...
0020100408700030500000...
https://www.sudokuwiki.org/...
42wrNNnlKjpM2_DggGmchv...
https://www.sudokuwiki.org/...
https://www.sudokuwiki.org/... ✕

Sudoku.wiki URL

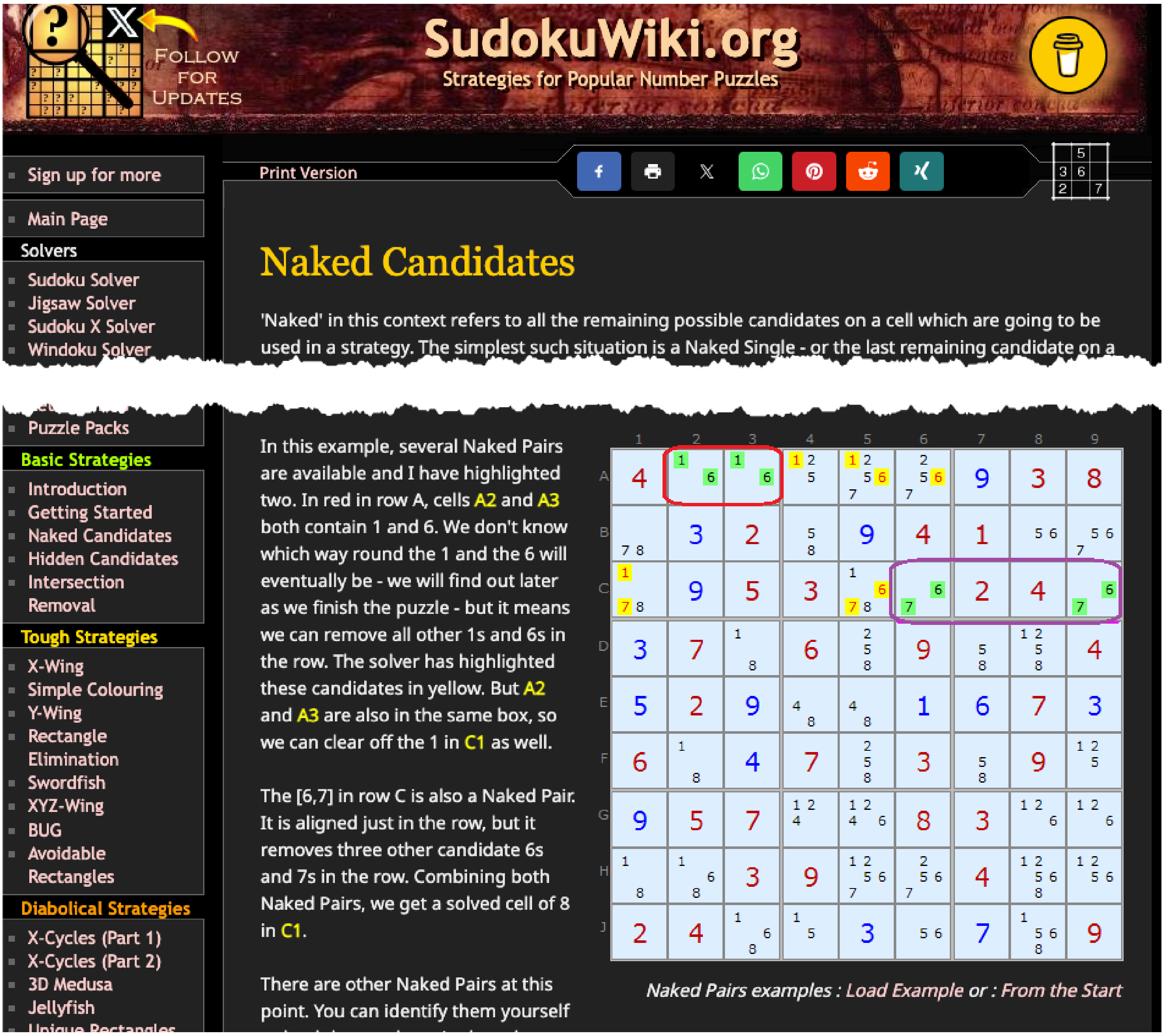
2				7			3	8
					6		7	
3				4		6		
		8		2		7		
1								6
		7		3		4		
		4		8				9
	6		4					
9	1			6				2

算法策略

给出良好的OOAD，能够高效、正确地在**数独乐乐**中引入、集成数独Wiki的各种算法策略。重点在于，

\1. 如何验证策略算法正确性；

\2. 如何在确保算法独立性，即新算法的引入不会影响老算法的正确性，也不会降低老算法的与性能。



SudokuWiki.org

Strategies for Popular Number Puzzles

Follow for Updates

- Sign up for more
- Main Page

Solvers

- Sudoku Solver
- Jigsaw Solver
- Sudoku X Solver
- Windoku Solver

- Puzzle Packs

Basic Strategies

- Introduction
- Getting Started
- Naked Candidates
- Hidden Candidates
- Intersection Removal

Tough Strategies

- X-Wing
- Simple Colouring
- Y-Wing
- Rectangle Elimination
- Swordfish
- XYZ-Wing
- BUG
- Avoidable Rectangles

Diabolical Strategies

- X-Cycles (Part 1)
- X-Cycles (Part 2)
- 3D Medusa
- Jellyfish
- Unique Rectangles

Print Version

Naked Candidates

'Naked' in this context refers to all the remaining possible candidates on a cell which are going to be used in a strategy. The simplest such situation is a Naked Single - or the last remaining candidate on a

In this example, several Naked Pairs are available and I have highlighted two. In red in row A, cells A2 and A3 both contain 1 and 6. We don't know which way round the 1 and the 6 will eventually be - we will find out later as we finish the puzzle - but it means we can remove all other 1s and 6s in the row. The solver has highlighted these candidates in yellow. But A2 and A3 are also in the same box, so we can clear off the 1 in C1 as well.

The [6,7] in row C is also a Naked Pair. It is aligned just in the row, but it removes three other candidate 6s and 7s in the row. Combining both Naked Pairs, we get a solved cell of 8 in C1.

	1	2	3	4	5	6	7	8	9
A	4	1 6	1 6	1 2 5	1 2 5 6 7	2 5 6 7	9	3	8
B	7 8	3	2	5 8	9	4	1	5 6	5 6 7
C	1 7 8	9	5	3	1 6 7 8	6 7	2	4	6 7
D	3	7	1 8	6	2 5 8	9	5 8	1 2 5 8	4
E	5	2	9	4 8	4 8	1	6	7	3
F	6	1 8	4	7	2 5 8	3	5 8	9	1 2 5
G	9	5	7	1 2 4	1 2 4 6	8	3	1 2 6	1 2 6
H	1 8	1 6 8	3	9	1 2 5 6 7	2 5 6 7	4	1 2 5 6 8	1 2 5 6
J	2	4	1 6 8	1 5	3	5 6	7	1 5 6 8	9

Naked Pairs examples : Load Example or : From the Start

There are other Naked Pairs at this point. You can identify them yourself