

# Анализ игры “Судоку”

# История sudoku

- Прообразом sudoku - "Магический квадрат", появился в Китае 2000 лет назад.
- Квадрат размером 3x3 клетки. В каждую клетку число от 1 до 9. Сумма чисел в любом столбце, строке и по диагонали равнялась 15.
- Леонард Эйлер(1707-1783). Магические квадраты для 9, 16, 25 и 36 ячеек.
- 1979г. Word Games magazine. Гарвард Гаррис.
- Su – число, duko – единственное.
- 12 ноября 2004г. The Times.

# Правила и цель игры

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

Квадрат 9x9. Цифры от 1 до 9. В строке столбце и малом квадрате 3x3 цифры не повторяются.

# Правила и цель игры

- Несколько уровней сложности(сложность зависит не только от количества заполненных клеток, а и от положения).
- Разные по размеру(4x4, 9x9,16x16 и так далее. В журналах встречаются нестандартного размера, пр. 9x12).
- Приёмы решения одинаковы(не зависят от размера и сложности, так что один и тот же алгоритм подойдёт как для 9x9, так и для 16x16).

# Способы решения sudoku

- Как задачи CSP
- Множество переменных,  $X_1, X_2, \dots, X_n$  и множеством ограничений,  $C_1, C_2, \dots, C_m$ . Каждая  $X_i$  имеет непустую область определения  $D_i$  (числа от 1 до 9) возможных значений.
- Каждое  $C_i$  включает некоторое подмножество переменных и задаёт допустимые комбинации значений для этого подмножества. Состояние задачи - присваивание значений некоторым переменным.
- Присваивание, которое не нарушает никаких ограничений, называется совместимым, или допустимым присваиванием.
- Полным называется такое присваивание, в котором участвует каждая переменная, а решением CSP задачи является полное присваивание, которое удовлетворяет всем ограничениям.

# Способы решения sudoku

- Любой задаче CSP может быть дана инкрементная формулировка, как и любой стандартной задаче поиска:
  - Начальное состояние. Изначально в некоторых клетках вписаны числа, то есть переменные, соответствующие этим клеткам, инициализированы некоторыми значениями, не нарушающими ограничения.
  - Функция определения преемника. Значение может быть присвоено любой переменной с неприсвоенным значением, при условии, что переменная не будет конфликтовать с другими переменными, значения которым были присвоены ранее.
  - Проверка цели. Проверка того, что текущее присваивание является полным, то есть заполнены все 81 клетки и не нарушается ни одно из ограничений.

# Способы решения sudoku

- Методы решения таких задач:
  - Поиск в ширину ( $n! \cdot d^n$  ветвей дерева)
  - Поиск в глубину
- Поиск в глубину может быть усовершенствован:
  - Эвристики (Minimum Remaining Values - MRV)
  - Предварительная проверка (распространение ограничений)
  - Хронологический поиск с возвратами (при неудаче возврат к предыдущей переменной, либо к переменной из конфликтного множества. Его определяет предварительная проверка)

# Приёмы решения sudoku

- Приёмы, упрощающие дерево поиска
  - Одиночки
  - Скрытые одиночки
  - Запертый кандидат
  - Открытые пары(тройки, четвёрки)
  - Скрытые пары(тройки, четвёрки)
  - X-wing
- Одиночки. Метод заключается в отыскании в таблице одиночек, т.е. ячеек, в которых возможна только одна цифра и никакая другая. Вписываем её и исключаем её из других клеток этой строки, таблицы и блока.

# Приёмы решения sudoku

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

Скрытые одиночки. 4 в зелёном блоке только в одной клетке.

# Приёмы решения sudoku

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

Запертый кандидат. В пределах блока цифра только в одной строке. Из других блоков исключаем.

# Приёмы решения sudoku

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

Открытые пары. Если две ячейки в группе (строке, столбце, блоке) содержат идентичную пару кандидатов и ничего более, то никакие другие ячейки этой группы не могут иметь значения этой пары.

# Приёмы решения sudoku

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

Скрытые пары. Если в двух ячейках в группе (строке, столбце, блоке) содержат кандидаты, среди которых идентичная пара, не встречающаяся ни в одной другой ячейке данного блока, то никакие другие ячейки этой группы не могут иметь значения этой пары.

# Приёмы решения sudoku

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

X-wing. Если значение имеет только два возможных местоположения в какой-то строке (столбце), то оно обязательно должно быть назначено в одну из этих ячеек. Если же существует еще одна строка (столбец), где этот же кандидат также может быть только в двух ячейках и столбцы (строки) этих ячеек совпадают, то ни одна другая ячейка этих столбцов (строк) не может содержать данную цифру.

# X-wing

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

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

# Реализация программы

- Среда разработки – Delphi 7.
- 3 режима :
  - Решение вводимого sudoku программой
  - Решение sudoku, сгенерированного программой, пользователем
  - Генерация sudoku, определённого уровня сложности
- Работает с sudoku общего вида, однако тестировалась в основном для 9x9.

# Реализация программы

- Решение вводимого sudoku программой.
- Программа генерирует пустой квадрат с количеством ячеек, определяемым пользователем.
- Пользователь может кликнуть на любую из ячеек и вписать туда цифру, нажатием соответствующей клавиши на клавиатуре.
- После введения некоторого количества чисел, пользователь может кликнуть по кнопке “Решить” и программа предоставит ему возможный вариант решения, либо сообщит о том, решение данного sudoku не существует.

# Реализация программы

- Введенный sudoku программа решает с помощью
  - Простой перебор с возвратами с эвристикой типа MRV (то есть вначале заполняются такие клетки, в которые можно подставить наименьшее количество цифр)
  - Второй алгоритм представляет собой тот же поиск с возвратами, однако в нём программа пытается найти ситуации, описанные ранее в “приёмах решения sudoku”.
- Программа выдаёт количество присваиваний, которое сделала в процессе решения, таким образом предоставляя возможность сравнить эффективность второго алгоритма, по сравнению с первым.
- Так же программа может выдавать количество возможных решений введенного sudoku.

# Реализация программы

- Решение sudoku, сгенерированного программой, пользователем.
- После выбора уровня сложности и количества ячеек, программа создаёт квадрат  $N \times N$ , некоторые числа которого заполнены цифрами.
- Пользователь, кликая по ячейкам и нажимая на цифры на клавиатуре, заполняет данный квадрат до того момента, пока остаются пустые клетки.
- В случае введения ошибочной цифры, пользователь может выбрать эту же клетку и поставить в неё другую цифру.

# Реализация программы

- Режим контроля ввода. При включении этой опции, программа будет выдавать ошибку в случаях, когда введённая пользователем цифра недопустима из-за того, что по горизонтали, вертикали или в блоке 3x3 такая цифра уже имеется.
- Режим подсказок. В случаях, когда пользователь затрудняется в выборе того, какую цифру и в какую клетку поставить, он может включить режим подсказок. В этом режиме, программа будет выдавать для каждой клетки перечень цифр, которые возможно поставить в эту клетку, без нарушения правил игры.

# Реализация программы

- Генерация sudoku выбранного уровня сложности.
- Модификация готового sudoku.
- Генерация ответа с последующим вычёркиванием некоторых цифр с проверкой единственности решения на каждом этапе.
- Заполнение пустого квадрата цифрами, с проверкой существования и единственности решения.

# Итоги и цели

- Генерация sudoku выбранного уровня сложности
- Тестирование работы алгоритмов решения для sudoku произвольного размера
- Поиск новых приёмов решения sudoku
- Задача sudoku, имеющего единственное решение при минимальном количестве цифр.
- <http://units.maths.uwa.edu.au/~gordon/sudokumin.php>

# Итоги и цели

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