ЛР з розділу 4, Щербакова Валерія, ФІ-71

3. Судоку

Сформулювати умови головоломки судоку як задачу пошуку розмітки на напівкільці $\langle\{0,1\},\vee,\wedge\rangle$ та реалізувати алгоритм її розв'язку.

1. На вхід програма приймає табличку з цифрами від 1 до 9 в заповнених клітинках та 0 в якості пустих клітинок.

2. На вихід програма виводить або розв'язок задачі (табличку з цифрами), або повідомлення про те, що розв'язок не було знайдено.

3. Необов'язковою корисною можливістю буде вивід на екран проміжних результатів: табличок, що містять цифри там, де вже обрано відповідь, та пробіли там, де ще є кілька варіантів можливих цифр.

Сформулирую условие головоломки судоку как CSP:

$Z = \langle T, \tau \subseteq T^2, K, g: \tau \times K^2 \to \{0,1\}\rangle$

Множество меток: $K = \{1,2,3,4,5,6,7,8,9\}$

Множество объектов $T = \{\forall i,j \in I: t_{ij}\}$ (здесь множество индексов $I = K$).

Условие соседства: $\tau\left(t_{ij}, t_{ab}\right) = (\llbracket i \neq a \rrbracket \wedge \llbracket j = b \rrbracket) \vee (\llbracket i = a \rrbracket \wedge \llbracket j \neq b \rrbracket) \vee$
$\vee (\llbracket i \neq a \rrbracket \wedge \llbracket j \neq b \rrbracket \wedge \llbracket s_i = s_a \rrbracket \wedge \llbracket s_j = s_b \rrbracket)$

Пусть множество соседей x_{ij}: $N_{ij} = \left\{\forall a,b \in I,\ \tau\left(t_{ij}, t_{ab}\right) = 1: \boldsymbol{t_{ab}}\right\}$

Вспомогательная функция s_i определяет квадрат:

$$\mathrm{s}_i = \lceil (i-1)/3 \rceil \overset{i \in I}{=} \begin{cases} 0, 1 \leq i \leq 3 \\ 1, 4 \leq i \leq 6 \\ 2, 7 \leq i \leq 9 \end{cases} = \llbracket i \geq 4 \rrbracket + \llbracket i \geq 7 \rrbracket.$$

s_i, s_j	i=1	2	3	4	5	6	7	8	9
j=1	1,1	1,1	1,1	2,1	2,1	2,1	3,1	3,1	3,1
2	1,1	1,1	1,1	2,1	2,1	2,1	3,1	3,1	3,1
3	1,1	1,1	1,1	2,1	2,1	2,1	3,1	3,1	3,1
4	1,2	1,2	1,2	2,2	2,2	2,2	3,2	3,2	3,2
5	1,2	1,2	1,2	2,2	2,2	2,2	3,2	3,2	3,2
6	1,2	1,2	1,2	2,2	2,2	2,2	3,2	3,2	3,2
7	1,3	1,3	1,3	2,3	2,3	2,3	3,3	3,3	3,3
8	1,3	1,3	1,3	2,3	2,3	2,3	3,3	3,3	3,3
9	1,3	1,3	1,3	2,3	2,3	2,3	3,3	3,3	3,3

Таким образом, соседями какого-то элемента x_{ij} считаются все элементы на одной вертикали с ним ($\llbracket i = a \rrbracket$), все элементы на одной горизонтали с ним ($\llbracket j = b \rrbracket$) а также все элементы в одном квадрате ($\llbracket s_i = s_a \rrbracket \wedge \llbracket s_j = s_b \rrbracket$), за исключением x_{ij}.

Ограничение простое – метки у соседей не должны совпадать: $g_{tt'}(k, k') = \llbracket k \neq k' \rrbracket$.

Вспомогательная функция индикатора установленной метки: $q_{ij}(x) = \llbracket k_{ij} = x \rrbracket$

Теперь следует описание непосредственно алгоритма решения задачи.

На вход передается $\{v_{ij}\} = v \in \widetilde{K}^{9\times 9}$, где $\widetilde{K} = K \cup \{0\}$, где 0 – не заполнена.

1. Задать метки объектов t_{ij} входными данными: $k_{ij} \coloneqq v_{ij}$.

2. Проверить $g_{t_{ij}t_{ab}}(k_{ij}, k_{ab})$ для $\forall k_{ij} \neq 0, \forall k_{ab} \in N_{ij}$. Если хотя бы одно значение g будет равно нулю, завершить алгоритм с результатом – задача не имеет решения по причине некорректного условия задачи.

3. Задать множество дополнительных ограничений – недопустимых меток для каждой ячейки $f \coloneqq \{\forall t_{ij}\colon f_{ij} = \emptyset\}$, $f \in (2^K)^{9\times 9}$.

4. Инициализировать переменную состояния для возврата меток, допустимых и запрещенных меток: $p \in \left(\widetilde{K}^{9\times 9} \times (2^K)^{9\times 9} \times (2^K)^{9\times 9}\right) \cup \{\otimes\}$ пустым значением (отсутствующего объекта) $p \coloneqq \otimes$ и текущий перебираемый объект $q = t_{00}$.

5. Определить допустимые метки для каждого объекта с $k_{ij} = 0$:

$$c_{ij} \coloneqq K \setminus \left(f_{ij} \cup \bigcup_{\substack{t_{ab} \in N_{ij} \\ k_{ab} \neq 0}} \{k_{ab}\} \right)$$

6. Если существует такой объект, для которого допустимо только одно значение метки $\exists c_{ij}\colon |c_{ij}| = 1$, то, обозначив $c_{ij} = \{d_{ij}\}$, зафиксировать это допустимое значение в разметке: $k_{ij} \coloneqq d_{ij}$ и перейти на шаг 5.

7. Если существует объект без допустимых значений: $\exists k_{ij} = 0, |c_{ij}| = 0$, то перейти к шагу 7.1 (продолжить выполнение), иначе перейти к шагу 8.

7.1. Если состояние меток и запретов для возврата не задано $p = \otimes$, то завершить алгоритм с результатом – задача не имеет решения по причине некорректного условия задачи (в процессе решения обнаруживаются противоречащие ограничения).

7.2. Вернуть значение переменных к ним $\langle k, f, c\rangle \coloneqq p$.

7.3. Определить следующее значение метки $r \coloneqq \min(c_{ij})$ для q. Если такое значение определить невозможно ($c_{ij} = \emptyset$), то завершить работу алгоритма с ответом, что алгоритм не может решить задачу.

7.4. Запретить это значение метки: для $q\colon f_{ij} \coloneqq f_{ij} \cup \{r\}$ и сохранить текущее значение переменных для возврата: $p \coloneqq \langle k, f, c\rangle$, после чего задать метку $k_{ij} \coloneqq r$ и перейти к шагу 5.

8. Если $\nexists k_{ij} = 0$, то завершить работу алгоритма с решением (ответом) k.

9. Выбрать первый незаполненный $q = t_{ij}\colon k_{ij} = 0$ и перейти к шагу 7.3.