۱ برنامه‌ریزی کلاس‌ها

الف : دامنه و محدودیت متغییرها

دامنه متغییرها

طبق صورت سوال می‌توان کلاس‌ها را در یکی از سه روز هفته برگزار کرد.(روزهای هفته را متناظر با مقدار $0, 1, 2$ در نظر می‌گیریم.) و برای هر کلاس هم یکی از اساتیدی که با توجه به محدودیت به سوال می‌توانند درس را ارائه کنند در نظر گرفته می‌شود.
دامنه هر یک از متغییرها به صورت زیر است:

$C_1 \in \{0, 1, 2\} \times \{C\}$
$C_1 \in \{(0, C), (1, C), (2, C)\}$

$C_2 \in \{0, 1, 2\} \times \{B, C\}$
$C_2 \in \{(0, B), (1, B), (2, B), (0, C), (1, C), (2, C)\}$

$C_3 \in \{0, 1, 2\} \times \{A, B, C\}$
$C_3 \in \{(0, A), (1, A), (2, A), (0, B), (1, B), (2, B), (0, C), (1, C), (2, C)\}$

$C_4 \in \{0, 1, 2\} \times \{A, B, C\}$
$C_4 \in \{(0, A), (1, A), (2, A), (0, B), (1, B), (2, B), (0, C), (1, C), (2, C)\}$

$C_5 \in \{0, 1, 2\} \times \{B, C\}$
$C_5 \in \{(0, B, (1, B), (2, B), (0, C), (1, C), (2, C)\}$

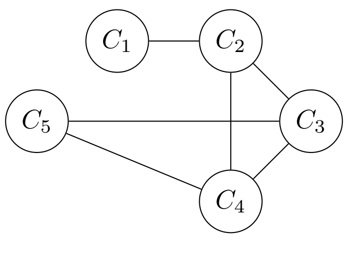
محدودیت‌های متغییرها

برای دو به دو کلاس‌هایی که با هم تلاقی زمانی دارند نمی‌توان یک استاد و یک روز مشخص را درنظر گرفت.
با توجه به صورت سوال و زمان‌های ذکر شده محدودیت‌ها به صورت زیر هستند:

$C_1 \neq C_2$
$C_2 \neq C_3$
$C_2 \neq C_4$
$C_3 \neq C_4$
$C_3 \neq C_5$
$C_4 \neq C_5$

ب : گراف محدودیت

راس‌ها : متغییرها(کلاس‌ها)
اگر دو متغییر در یک محدودیت حضور دارند یال رسم می‌کنیم.



پ : سازگاری یالی (arc consistency)

برای سازگاری یالی باید تمام روابط در نظر گرفته شود و اگر اخذ مقداری توسط یکی از متغییرها باعث شود دامنه مجاز متغییر دیگر تهی شود این مقدار دهی معتبر نیست.

$C_1 \in \{(0, C), (1, C), (2, C)\}$
$C_2 \in \{(0, B), (1, B), (2, B), (0, C), (1, C), (2, C)\}$
$C_3 \in \{(0, A), (1, A), (2, A), (0, B), (1, B), (2, B), (0, C), (1, C), (2, C)\}$
$C_4 \in \{(0, A), (1, A), (2, A), (0, B), (1, B), (2, B), (0, C), (1, C), (2, C)\}$
$C_5 \in \{(0, B), (1, B), (2, B), (0, C), (1, C), (2, C)\}$

چون محدودیت‌های مسئله به صورت نامساوی هستند اگر متغییری یکی از مقادیر را احذ کند برای متغییر دیگر موجود در محدودیت هم در دامنه خود مقادیری برای انتخاب باقی می‌ماند پس این گراف arc consistent است و دامنه‌ها تغییر نکرد.

ت : راه حل

می‌توان این مسئله را با روش backtrack حل کرد.
به اینصورت که در هر مرحله یکی از متغییرها را مقداردهی می‌کنیم و به ترتیب مشخصی به سراغ متغییرها دیگر می‌رویم. اگر در هر عمقی از درخت جست‌وجو تعداد اعضای باقی‌مانده در دامنه هر یک از متغییرها صفر شد یعنی ادامه دادن این شاخه لازم نیست. پس عقب‌گرد انجام می‌شود و یک یا حتی چند مرحله می‌توان به عقب برگشت و در گره‌های با عمق کم‌تر مقدارهای مناسب‌تری انتخاب کرد.
به این ترتیب مسئله به صورت دقیق حل می‌شود و اگر جواب نداشته باشد ، حل‌ناپذیر بودن نمونه ثابت می‌شود.

ث : چرا باید CSP ساختار درخت‌گونه داشته باشد؟

چون اگر گراف وابستگی‌ها به صورت درخت باشد امکان حل مسئله با اعمال یک ترتیب مناسب و بررسی پایداری یالی Arc Consistency می‌توان در زمان چندجمله‌ای بسیار بهینه دامنه‌ها متغییرها را به نوعی اصلاح نمود که اگر از گره ریشه پیمایش کنیم در کل فرایند مقداردهی به متغییرها نیاز به بازگشت به عقب و تصحیح مقداردهی‌های قبلی وجود نخواهد داشت.
اگر گراف محدودیت‌های مسئله درخت نباشد ولی تعداد دورها و یال‌های اضافه بسیار را شامل نشود می‌توان با روش‌هایی مانند تجزیه درختی مسئله را به حالت درختی تبدیل کرد تا از مرتبه زمانی حل نمایی به چند جمله‌ای تبدیل شود. ولی اگر تعداد یال‌های اضافه نسبت به درخت یافتن تجزیه و تبدیل به حالت درختی خود پروسه‌ای زمان‌بر و هزینه‌بر خواهد بود و تفاوت خاصی در حل مسئله ایجاد نخواهد شد.

ج : روش‌های تبدیل ساختار نیمه درخت به درخت

حذف راس‌ها : در این روش باید برخی رئوس گراف حذف شوند به طوری که گراف مانده یک درخت باشد.برای حل مسئله ارضا محدودیت با این روش لازم است به متغییرهای متناظر با راس‌های حذف شده مقداردهی شود و برای هر حالت مقدار مسئله‌ی متناظر با درخت به صورت مستقل حل شود. در این‌صورت اگر تعداد راس‌های حذف شده زیاد باشد تعداد مقداردهی‌ها بسیار زیاد خواهد بود. پس زمان حل هم بسیار طولانی خواهد شد.

چ : پیاده‌سازی با MiniZinc

مسئله فوق در این فایل آمده است.

۲ دنباله‌ی اعداد

الف : تعداد مقداردهی‌های ارضاکننده

برای این مسئله فقط یک مقداردهی ارضاکننده وجود دارد.چون اعداد باید اکیدا صعودی باشند و مرتب‌سازی اعداد یک آرایه به صورت یکتا انجام می‌شود پس باید مقدارها را به ترتیب از ۱ تا ۱۰ به متغییرها اختصاص بدهیم.

$x_1 = 1$
$x_2 = 2$
$x_3 = 3$
$x_4 = 4$
$x_5 = 5$
$x_6 = 6$
$x_7 = 7$
$x_8 = 8$
$x_9 = 9$
$x_{10} = 10$

ب : اصلاح دامنه x_1 وقتی نسبت به x_2 **Arc consistent** باشد

در این حالت اگر متغییر x_1 مقدار حداکثر (در این مقدار دهی 10) را اختیار کند متغییر x_2 هیچ یک از مقادیر دامنه خود را نمی‌تواند اختیار کند پس باید عدد 10 را از دامنه x_1 حذف کنیم.

$D_{x_1}^{new} = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$

پ : x_2 نسبت به x_3 و x_3 نسبت به x_4 دارای Arc Consistency باشد. برای Arc Consistency بین ۱ و ۲ چه باید کرد؟

دامنه‌های متغییرهای x_3 و x_4 را تعیین می‌کنیم.

$D_{x_1} = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$
$D_{x_2} = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$
$D_{x_3} = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$
$D_{x_4} = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$

در حالتی که در مسئله ذکر شده است دامنه به صورت فوق است حال می‌خواهیم پایداری یالی بین ۱ و ۲ ایجاد کنیم لازم است مقادیری از دامنه متغییر ۱ که باعث می‌شود دامنه مجاز متغییر ۲ تهی شود را حذف کنیم پس لازم است. متغییرهای ۹ و ۱۰ حذف شوند پس در نهایت دامنه‌های نهایی متغییرها ذکر شده پس از برقراری پایداری یالی بین ۱ و ۲ به صورت زیر خواهد بود.

$D_{x_1} = \{1, 2, 3, 4, 5, 6, 7, 8\}$
$D_{x_2} = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$
$D_{x_3} = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$
$D_{x_4} = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$

ت : وقتی x_i و x_{i+1} به ازای ($2 \leq i \leq N-1$) پایداری یالی داشته باشد. برای پایدار شدن x_1 و x_2 دامنه x_1 چه تغییری می‌کند؟

دامنه‌ی متغییرها به صورت زیر است.

$D_{x_1} = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$
$D_{x_2} = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$
$D_{x_3} = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$
$D_{x_4} = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$
$D_{x_5} = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$
$D_{x_6} = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$
$D_{x_7} = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$
$D_{x_8} = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$
$D_{x_9} = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$
$D_{x_{10}} = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$

اگر پایداری یالی بین x_i و x_{i+1} به ازای ($2 \leq i \leq N-1$) برقرار شود. دامنه‌ها به صورت زیر خواهد بود.

$D_{x_1} = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$
$D_{x_2} = \{1, 2\}$
$D_{x_3} = \{1, 2, 3\}$
$D_{x_4} = \{1, 2, 3, 4\}$
$D_{x_5} = \{1, 2, 3, 4, 5\}$
$D_{x_6} = \{1, 2, 3, 4, 5, 6\}$
$D_{x_7} = \{1, 2, 3, 4, 5, 6, 7\}$
$D_{x_8} = \{1, 2, 3, 4, 5, 6, 7, 8\}$
$D_{x_9} = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$
$D_{x_{10}} = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$

ترتیب پایدار کردن در مثال فوق از متغیرهای با شماره بیش‌تر به متغیر با شماره‌های کم‌تر انجام شده است.
حال می‌خواهیم متغییر x_1 را نسبت به x_2 اصلاح کنیم لازم است دامنه متغییر x_1 تغییر کند.
اگر متغییر x_1 هر یک از مقادیر بیش از ۱ را اخذ کند. ($x_1 > 1$) پس برای برقراری محدودیت‌های ذکر شده در مسئله لازم است. $x_2 > 1$ باشد پس با توجه به دامنه x_1 هیچ مقداری قابل انتخاب نخواهد بود.پس با توجه به استدلال فوق تمام مقادیر بیش‌تر از ۱ از دامنه x_1 آن حذف می‌شوند و دامنه‌ها در نهایت به صورت زیر خواهد بود.

$D_{x_1} = \{1\}$
$D_{x_2} = \{1, 2\}$
$D_{x_3} = \{1, 2, 3\}$
$D_{x_4} = \{1, 2, 3, 4\}$
$D_{x_5} = \{1, 2, 3, 4, 5\}$
$D_{x_6} = \{1, 2, 3, 4, 5, 6\}$
$D_{x_7} = \{1, 2, 3, 4, 5, 6, 7\}$
$D_{x_8} = \{1, 2, 3, 4, 5, 6, 7, 8\}$
$D_{x_9} = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$
$D_{x_{10}} = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$

ث : حداقل تعداد یال پردازش شده در الگوریتم AC-3

در این الگوریتم لازم است همه‌ی یال‌ها بررسی شوند تا پایداری یالی ثابت شود. پس به تعداد یال‌های موجود در درخت فوق یعنی $N - 1$ یال (رابطه یا محدودیت بین متغییرها) باید بررسی شود.

ج : مسئله با فرض صعودی یا نزولی بودن

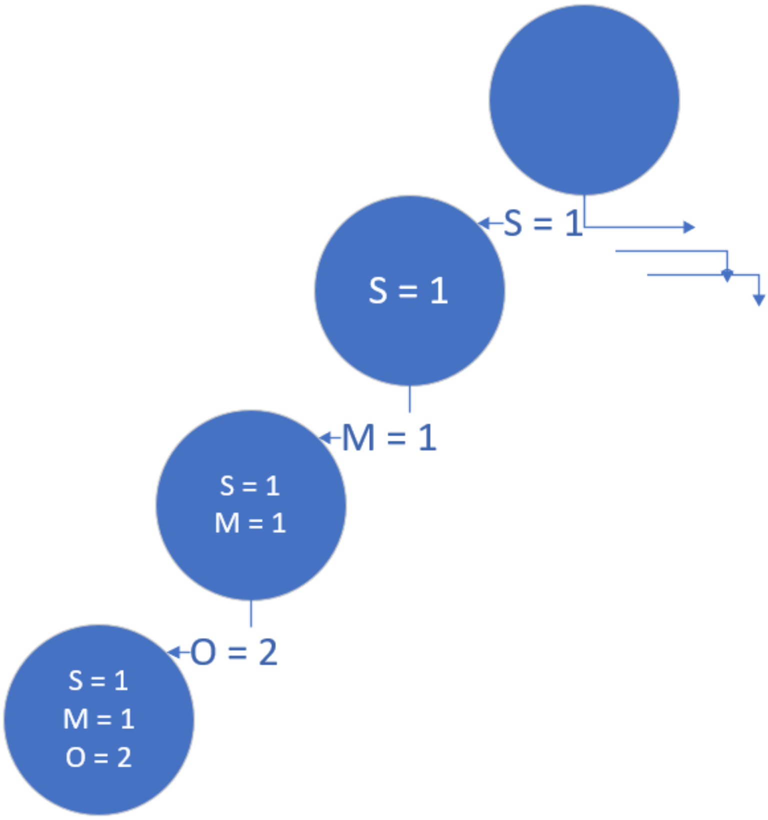
برای مسئله جدید علاوه بر متغییرهای قبلی یک متغییر بیتی جدید به نام s در نظر می‌گیریم به اینصورت که اگر این متغییر غلط باشد دنباله نزولی و اگر درست باشد دنباله صعودی است. پس دامنه‌ها و محدودیت‌ها در مسئله جدید به صورت زیر خواهند بود. (با فرض $N = M = 3$)

Variables:
$D_s = \{true, false\}$
$D_{x_1} = \{1, 2, 3\}$
$D_{x_2} = \{1, 2, 3\}$
$D_{x_3} = \{1, 2, 3\}$

Constraints:
$(s = true) \vee (x_1 > x_2 > x_3)$
$(s = false) \vee (x_1 < x_2 < x_3)$

۳ پازل کلمات

در توابع ابتکاری ذکر شده به این صورت است که در هنگام انتخاب متغییر برای مقداردهی ، متغییری انتخاب می‌شود که کم‌ترین تعداد اعضای دامنه مجاز را دارد. مثلا متغییری که یک عضو در دامنه خود دارد انتخاب می‌شود و باعث می‌شود تا در صورت نامعتبر بودن و نیاز به بازگشت به عقب سریع‌تر و در عمق کم‌تری این اتفاق رخ دهد.
حال برای اختصاص یک مقدار به متغییر باید مقداری اختصاص پیدا کند که محدودیت کم‌تری با دیگران دارد بیش‌ترین مقادیر را در دامنه‌های متغییرهای باقی‌مانده نتیجه می‌دهد.



شکل ۱: قسمتی از گراف مقداردهی به متغییرها

با توجه به گراف در ابتدا هیچ یک از متغییرها انتخاب نشده‌اند بعد از آن متغییری که کوچک‌ترین اندازه دامنه را دارد انتخاب می‌شود (چون صفر جزو دامنه متغییر های s ، m نیست) متغییرهای s ، m دامنه‌ی کوچک‌تری دارند و در نتیجه ابتدا برای مقداردهی انتخاب می‌شوند و از بین اعضای موجود در دامنه مقدارهای انتخاب می‌شوند که شروط کم‌تری نقض شوند مثلا متغیر o در انتخاب مقدار سعی شده است که حداقل با متغییرهای قبلی سازگار باشد و تلاقی کم‌تری ایجاد شود.

۴ سودوکو ۴ تایی

الف : متغییرها و محدودیت‌ها

برای هر خانه مجهول یک متغییر و برای هر سطر و ستون و قطر باید مقادیر متفاوت باشند.

Variables : A, B, C, D, E, F, G

Constraints :
Rows:
$AllDifferent\,\{2, A, 3, B\}$
$AllDifferent\,\{4, C, 1, 2\}$
$AllDifferent\,\{1, D, E, F\}$
$AllDifferent\,\{3, G, 4, 1\}$
Columns:

$AllDifferent\,\{2,4,1,3\}$
$AllDifferent\,\{A,C,D,G\}$
$AllDifferent\,\{1,D,E,F\}$
$AllDifferent\,\{B,2,F,1\}$
Diameters:
$AllDifferent\,\{2,C,E,1\}$
$AllDifferent\,\{3,D,1,B\}$

متغییر	A	B	C	D	E	F	G
مقادیر باقیمانده	$1,4$	4	3	$2,3,4$	2	$3,4$	2
دارای محدودیت با ... متغییر دیگر	4	2	3	5	2	3	3

پ : متغییرهای انتخابی بعدی با الگوریتم MRV (Min Remaining Value)

متغییرهایی انتخاب می‌شوند که کم‌ترین تعداد اعضای باقی‌مانده در دامنه را داشته باشند. در این مثال متغییر B,C,E,G ممکن است انتخاب شوند و انتخاب آن‌ها فقط مقداردهی است و هیچ تصمیم‌گیری لازم ندارد.
پس از آن متغییری که کم تر از بقیه مقدار باقی مانده دارد یعنی A,F انتخاب می‌شود و پس از آن نوبت به D می‌رسد.
$B \to C \to E \to G \to A \to F \to D$

ت : متغییرهای انتخابی بعدی با الگوریتم Degree

در این روش متغییری که درگیری بیش‌تری با سایر متغییرها دارد انتخاب می‌شود. پس در این مثال متغییر D انتخاب و مقداردهی می‌شود.

ث : روش بررسی پیش‌رو

Constraint propagation	A	B	C	D	E	F	G
All values	$1,4$	4	3	$2,3,4$	2	$3,4$	2
Constraint A , B	1	4	3	$2,3,4$	2	$3,4$	2
Constraint F , B	1	4	3	$2,3,4$	2	3	2
Constraint D , C	1	4	3	$2,4$	2	3	2
Constraint D , G	1	4	3	4	2	3	2

۵ مربع جادویی

مقدار مجموع ثابت سطر و ستون

با توجه به رابطه جمع اعداد طبیعی متوالی خواهیم داشت.
Sum of whole table $= 1+2+3+\cdots+n^2 = \frac{(n^2)\times(n^2+1)}{2}$
Sum every row , column or diameter $= \frac{1+2+3+\cdots+n^2}{n} = \frac{(n^2)\times(n^2+1)}{2n} = \frac{(n)\times(n^2+1)}{2}$

مدل‌سازی به صورت مسئله CSP

هر یک از خانه‌های جدول را یک متغییر در نظر می‌گیریم.($x_{i,j}$)
دامنه‌ی متغییرها اعداد طبیعی از 1 تا n^2 است.($D_{x_{i,j}} = \{a | a \in \{1,2,3,\cdots,n^2\}\}$)
محدودیت‌ها را به صورت قیدهای $AllDifferent$ نمایش می‌دهیم به طوریکه در هر سطر و ستون و قطر باید اعداد متفاوت باشند.

Row constraints $\forall i\ AllDifferent\left(x_{i,1}, x_{i,2}, x_{i,3}, \cdots, x_{i,n}\right)$
$x_{i,1} + x_{i,2} + x_{i,3} + \cdots + x_{i,n} = \frac{(n)\times\left(n^2+1\right)}{2}$

Column constraints $\forall j\ AllDifferent\left(x_{1,j}, x_{2,j}, x_{3,j}, \cdots, x_{n,j}\right)$
$x_{1,j} + x_{2,j} + x_{3,j} + \cdots + x_{n,j} = \frac{(n)\times\left(n^2+1\right)}{2}$

Diameter constraints $AllDifferent\left(x_{1,1}, x_{2,2}, x_{3,3}, \cdots, x_{n,n}\right)$
$x_{1,1} + x_{2,2} + x_{3,3} + \cdots + x_{n,n} = \frac{(n)\times\left(n^2+1\right)}{2}$

$AllDifferent\left(x_{1,n}, x_{2,n-1}, x_{3,n-2}, \cdots, x_{n,1}\right)$
$x_{1,n} + x_{2,n-1} + x_{3,n-2} + \cdots + x_{n,1} = \frac{(n)\times\left(n^2+1\right)}{2}$

۶ تجزیه درختی

اندازه تجزیه درختی برابر 3 است. دسته‌های مربوط به تجزیه درختی به صورت زیر است.

$\{A, B, C, D\}$
$\{B, C, D, E\}$
$\{C, D, E, F\}$
$\{D, E, F, G\}$
$\{E, F, G, H\}$
$\{F, G, H, I\}$

درباره یک ساختار گرید $m * n$ می‌توان گفت که عرض درختی برابر $min\,\{m, n\}$ است. اگر گرید به صورت زیر شماره گذاری شده باشد.

$$\begin{array}{ccccccc}
1 & - & 2 & - & 3 & \dots & m \\
| & & | & & | & & | \\
m+1 & - & m+2 & - & m+3 & \dots & 2m \\
\vdots & - & \vdots & - & \vdots & \ddots & \vdots \\
m*(n-1)+1 & - & m*(n-1)+2 & - & m*(n-1)+3 & \dots & m*n
\end{array}$$

دسته‌های موجود در تجزیه درختی به صورت زیر خواهد بود.

$Bags = \{\{k, k+1, ..., k+m\}\, | k, \cdots, k+m \subseteq \{1, 2, 3, ..., min\,\{m, n\}\}\}$

۷ رنگ‌آمیزی گراف

الف : بررسی پایداری

Arc consistency

اگر مقداردهی معتبر برای یک متغیر دلخواه فرض کنیم آنگاه می‌توان برای هر متغیر دیگری حداقل یک مقدار معتبر ارائه کرد. پس پایداری یالی وجود دارد.

Path consistency

اگر مقداردهی معتبر برای دو متغیر دلخواه فرض کنیم آنگاه می‌توان برای هر متغیر دیگری حداقل یک مقدار معتبر ارائه کرد. پس پایداری سه‌تایی وجود دارد.

4 consistency

اگر مقداردهی معتبر برای سه متغیر دلخواه فرض کنیم چون گراف کامل است برای متغیر چهارم امکان انتخاب هیچ یک از رنگ‌های قبلی وجود ندارد پس پایداری از درجه ۴ وجود ندارد.

هم چنین با توجه به حل ناپذیر بودن مسئله امکان برقراری پایداری از درجه ۴ به طور کلی وجود ندارد و با تغییر هیچ‌یک از روابط ممکن نیست.

ب : R یک مسئله سه رنگ‌آمیزی است. برقراری پایداری از درجه ۲ و ۴

برای بررسی پایداری درجه دو یا همان پایداری یالی باید یک متغییر مقدار بگیرد و پس از آن امکان مقداردهی به هر متغییر دیگر که با آن رابطه دارند فراهم باشد و با توجه به در اختیار داشتن سه رنگ اگر یک متغییر مقداردهی شود باز هم برای هر متغییر دیگر امکان انتخاب حداقل یک گزینه وجود دارد پس این پایداری هیچ تغییری در دامنه‌ی متغییرها ایجاد نمی‌کند و تغییری در فضای جست‌وجو ایجاد نخواهد شد.
برای برقراری پایداری از درجه ۴ باید برای هر سه متغییر دلخواه مقداردهی با توجه به محدودیت‌های مسئله انجام شود.در صورتی که به ازای یک مقداردهی مشخص برای متغییر دیگری امکان انتخاب مقدار از دامنه نباشد باید با استفاده از یک رابطه جدید بین سه متغییر مقداردهی شده اولیه این مقداردهی مشخص را غیرمجاز کرد.(در صورتی که گراف یک زیرگراف کامل با اندازه ۴ داشته باشد امکان برقراری این پایداری وجود ندارد و مسئله بدون جواب است.)

۸ پیاده‌سازی

۱.۸ پیاده‌سازی مربع جادویی با ابزار OR-Tools

پیاده‌سازی این برنامه در پوشه کدها آمده است. فایل برنامه اصلی
روش اجرا : در دایرکتوری پروژه در ترمینال دستور mvn compile exec:java را وارد کنید. پس از آن برنامه اندازه مسئله را دریافت کرده و جواب به دست آمده را نشان می‌دهد.

```
Enter size of magic square : 20
398   386   3     17    400   1     4     391   399   396   21    393   2     9     394   381   7     5     397   6
3     399   398   395   396   292   400   10    106   350   9     2     1     4     56    6     397   7     386   393
6     235   400   398   394   4     2     399   392   5     3     1     18    12    391   8     393   395   396   158
11    398   397   8     393   49    90    2     3     400   7     6     390   392   395   399   268   396   5     1
395   7     10    392   3     399   5     269   398   4     400   394   396   1     397   14    129   2     6     389
2     175   6     64    399   107   3     4     393   296   8     396   5     398   332   7     400   226   392   397
4     394   7     10    391   6     396   392   2     397   398   3     389   8     5     1     395   393   400   19
396   400   35    5     8     397   365   1     395   7     2     4     399   394   393   153   6     249   398   3
393   392   399   323   395   394   7     3     35    46    6     5     400   10    398   2     13    389   4     396
391   2     33    7     9     398   226   6     1     222   396   397   3     400   11    395   4     394   399   316
394   1     5     397   79    7     399   297   4     398   393   395   391   11    2     392   3     6     36    400
8     4     393   9     2     3     398   400   397   6     399   391   394   7     396   5     1     392   395   10
400   396   340   391   398   2     6     393   5     1     4     399   397   395   7     3     9     8     394   62
9     5     2     393   397   395   1     394   7     399   387   400   4     6     10    396   398   12    3     392
386   397   1     3     6     400   119   395   394   282   5     8     9     396   92    398   399   4     2     314
1     6     396   400   190   5     395   397   382   393   394   398   7     399   9     4     2     3     30    199
397   393   395   2     4     396   10    5     6     3     392   7     398   365   8     400   394   399   1     35
10    3     4     291   5     8     397   7     400   2     1     120   393   391   259   394   396   268   262   399
399   9     394   396   140   392   393   236   281   12    390   10    8     15    400   259   11    72    84    109
7     8     392   109   1     355   394   9     10    391   395   281   6     397   55    393   385   390   20    12

Statistics
  - conflicts : 1276
  - branches  : 18262
  - wall time : 1.1660242570000001 s
  - solutions : 1
```

شکل ۲: نمونه‌ی اجرای برنامه مربع جادویی

۲.۸ پیاده‌سازی N وزیر با ابزار Jacop

فایل برنامه اصلی
روش اجرا با یک برنامه مثل Intellij idea پروژه را باز کرده و فایل را اجرا کنید. برنامه در ابتدای کار اندازه جدول را دریافت می‌کند و سپس دنباله اعداد متناظر با شماره ردیف هر وزیر که در ستون‌های مختلف قرار دارد را چاپ می‌کند.

```
Enter number of Queens : 200
Labeling has finished with return value of true
DFS1: DFS([Q1 = 100, Q2 = 98, Q5 = 99, Q7 = 102, Q12 = 101, Q16 = 10

Standard output : (space separated list of queen placements in colum
100 98 29 65 99 134 102 161 136 132 168 101 126 164 67 103 127 73 12

Execution time : 14151 milli seconds

Process finished with exit code 0
```

شکل ۳: اجرای برنامه n وزیر