

سودوکو اره‌ای

مقدمه

سودوکو، مخفف یک عبارت ژاپنی به معنی «ارقام باید تنها باشند» است. نوع متداول سودوکو یک جدول ۹ در ۹ است که کل جدول هم به ۹ جدول کوچکتر (در حالت رایج ۳ در ۳) ولی در اینجا -سودوکو اره‌ای- بدون محدودیت شکل و تنها محدود به بخش‌های ۹ تایی (تقسیم شده است. در این جدول چند عدد به طور پیش فرض قرار داده شده که باید باقی اعداد را با رعایت سه قانون زیر یافت

- قانون اول: در هر سطر جدول اعداد ۱ الی ۹ بدون تکرار قرار گیرد.
- قانون دوم: در هر ستون جدول اعداد ۱ الی ۹ بدون تکرار قرار گیرد.
- قانون سوم: در هر ناحیه جدول اعداد ۱ الی ۹ بدون تکرار قرار گیرد.

نمونه جدول

		۸					۹	
	۹		۶			۸		۴
۷				۹			۵	
	۲		۳					
		۶		۴		۳		
					۷		۴	
	۶			۳				۲
۳		۹			۸		۲	
	۸					۹		

		۶				۸		
	۵		۲		۹		۳	
۱				۹				۳
	۳		۹		۲		۴	
		۹		۷		۲		
	۷		۳		۵		۹	
۲				۶				۹
	۲		۶		۸		۷	
		۷				۹		

نمونه‌های بیشتر، توضیحات و راه حل‌های آنها را می‌توانید در این [لینک](#) دنبال کنید. بسیاری از مسائل مطرح در زمینه هوش مصنوعی را میتوان به صورت مسائل ارضای محدودیت توصیف کرد. این مسائل با استفاده از مجموعه‌ای از متغیرها و تعدادی محدودیت برای مقادیری که این متغیرها میتوانند اختیار کنند، تعریف میشوند. حل این مسائل مجموعه‌ای از مقادیر منحصر به فرد برای متغیرهاست، به طوری که تمامی محدودیت‌های موردنظر مسئله ارضا شده باشد. در این پروژه هدف حل جداول سودوکو اره‌ای با کمک گرفتن از الگوریتم‌های عقبگرد (۱) با سه هیوریستیک

کمترین مقادیر باقی مانده ۲، درجهای ۳ و مقدار با کمترین محدودیت ۴، سازگاری کمان ۵ - ۳ و بررسی رو به جلو ۶ است.

برنامه شما باید الگوریتمهای بالا را به صورت انتخابی داشته باشد و زمان حل مسئله را در هر حالت نشان دهد. ورودی برنامه به ازای هر جدول شامل ۲ فایل است که در یکی از آنها مقادیر جدول و در دیگری بخشبندی آن مشخص شده است.
برای مثال:

1	1	6	6	6	6	6	6	6	4	3	7	0	0	0	0	0	0	0
1	1	7	7	6	7	6	8	8	5	0	0	0	0	0	0	0	9	0
1	1	1	7	7	7	7	8	8	0	0	0	0	3	0	0	0	0	0
2	1	1	5	7	7	8	8	8	0	8	0	3	0	1	0	0	0	0
2	5	5	5	5	5	5	5	8	0	0	0	8	1	5	0	0	0	0
2	2	2	4	4	5	9	9	8	0	0	0	1	0	2	0	6	0	0
2	2	4	4	4	4	9	9	9	0	0	0	0	8	0	0	0	0	0
2	2	3	4	3	4	4	9	9	0	5	0	0	0	0	0	0	0	4
3	3	3	3	3	3	3	9	9	0	0	0	0	0	0	0	8	7	2

- 1 Backtracking
- 2 Minimum Remaining Value
- 3 Degree
- 4 Least Constraining Value
- 5 Arc Consistency
- 6 Forward Checking

- تسلط کامل به کد ارسالی لازم است.
- در صورت مشاهده هرگونه تقلب **نمره صفر** منظور میشود.
- تنها استفاده از کتابخانههای عام منظوره قابل قبول است. (تمامی الگوریتمها بایستی توسط شما پیاده‌سازی شود)
- در پیوست توضیحاتی کوتاه از الگوریتمهای مورد استفاده در پروژه آورده شده است.

پیوست:

الگوریتم عقبگرد:

شیوهای در حل مسائل است که از علامتهای خاصی برای بیان اینکه راه حل کاندیدی به حل مسئله میانجامد یا خیر استفاده میکند. این رویکرد برای حل مسائل درخت فضای آن مسئله را ایجاد کرده و تعیین میکند کدام گره امیدبخش است. عقبگرد حالت اصلاح شده جستجوی عمقی یک درخت است.

الگوریتم سازگاری کمان ۳:

الگوریتم یک صف از کمانها تشکیل میدهد

هر بار یک کمان $X \rightarrow Y$ از صف انتخاب شده و سازگاری کمان برای آن بررسی میشود

اگر دامنه X تغییری نکرد، به سراغ کمان بعدی میرود

اگر دامنه X تغییر کرد، کمانهای $X \rightarrow Z$ برای تمام همسایههای X را به صف اضافه میکند

اگر دامنه X تهی شد، مسئله جواب ندارد

الگوریتم بررسی رو به جلو:

وقتی مقداری به متغیر فرضی X داده میشود؛ فرآیند بررسی رو به جلو متغیرهای بدون انتساب مثل Y را در نظر می‌گیرد که از طریق یک محدودیت به X متصل است و هر مقداری که با مقدار انتخاب شده برای X همخوانی ندارد را از دامنه Y حذف میکند. اگر مقدار انتخاب شده برای X باعث شود دامنه یکی از متغیرها خالی شود؛ آن مقدار را به X نمی‌دهیم و اگر همه مقادیر X را بررسی کرده‌ایم عقبگرد میکنیم.

• برای توضیحات بیشتر در هر مورد می‌توانید به اسلایدهای درس و نمونه کدهای آن مراجعه کنید.