

پروژه سودوکو

علی مومنی

استاد : سمیرا لویمی

بهار ۱۴۰۱

نحوه فرموله کردن مسئله :

- متغیر ها :
- خانه های خالی : هدف پر کردن خانه های با مقدار صفر یا به اصلاح خانه های خالی است .

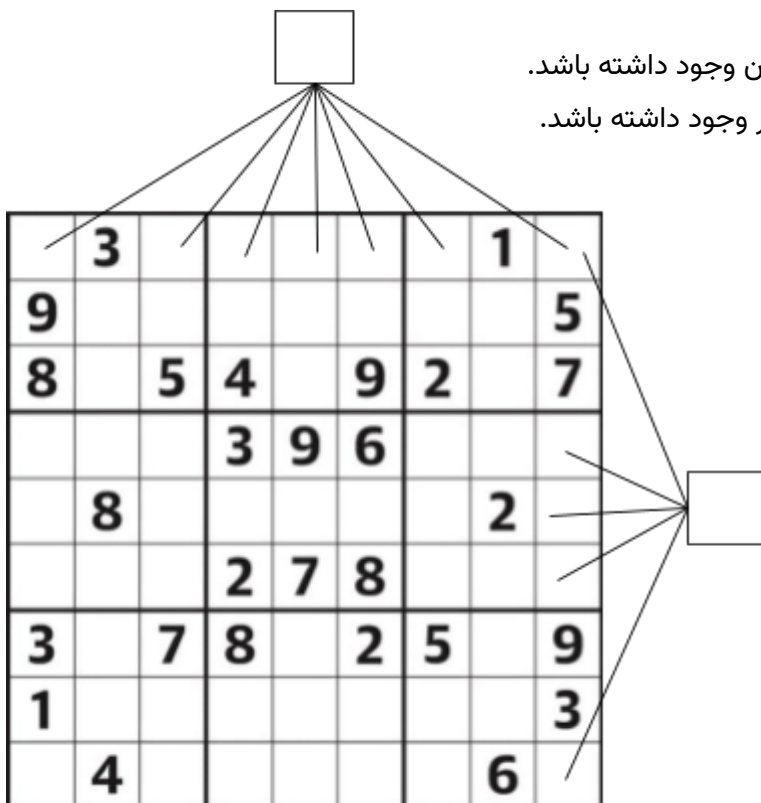
- دامنه ها: دامنه اعداد با توجه به سایز سودوکو قابل تغییر است که از ارقام 1 تا n تغییر میکند که n برابر سایز سودوکو است

- محدودیت های مسئله :
- N مقدار مختلف در هر ستون وجود داشته باشد.
- N مقدار مختلف در هر سطر وجود داشته باشد.

مثال برای سودوکو 9\*9 :

- متغیر ها :
- خانه های خالی : هدف پر کردن خانه های با مقدار صفر یا به اصلاح خانه های خالی است .
- دامنه ها: ارقام 1 تا 9

- محدودیت های مسئله :
- 9 مقدار مختلف در هر ستون وجود داشته باشد.
- 9 مقدار مختلف در هر سطر وجود داشته باشد.



## توضیحات کد ها:

### الگوریتم BT :

ابتدا همه خانه های که مقدار صفر دارند را به کمک کتابخانه numpy پیدا کرده آنها را در آرایه vars قرار میدادیم و هر بار که یکی از خانه ها پر میشود دوباره همین کار انجام میشود .

```
vars = [tuple(e) for e in np.transpose(np.where(sudoku==0))]
```

سپس با استفاده از هیوریستیک MRV و درجه ای مقادیر مربوط به هر خانه را به دست میاوریم در هیوریستیک MRV ما تعداد کل حالت های مینیمم را به دست می آوریم و بعد از آن ماکسیمم درجه هر کدام از آنها را به هیوریستیک درجه ای به دست می آوریم.

### الگوریتم FC :

بعد از مرحله اول با کمک گرفتن از الگوریتم FC مقادیری که قرار است انتخاب شوند را چک میکنیم اگر مقداری که چک میشود اوکی باشد الگوریتم FC مقدار true و اگر مقداری که چک میشود اوکی نباشد و با بقیه مقادیر جدول تداخل داشته باشد ( برابر باشد ) مقدار false را برمیگرداند این کار در هر مرحله برای مقادیر انجام میشود.

در آخر اگر خانه از جدول مقدار صفر نداشته باشد یعنی هر خانه در هر سطر و ستون فقط با اعداد 1 تا n پر شده باشد و در هر سطر و ستون هیچ عدد تکراری وجود نداشته باشد.