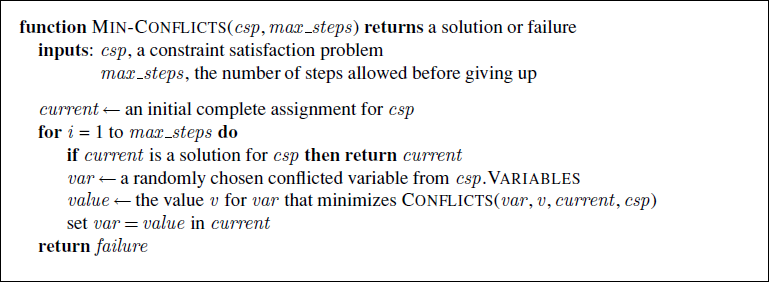
**مسئله هشت وزیر**

در این پروژه مسئله معروف قرار دادن هشت وزیر در صفحه شطرنج به طوری که هیچ یک به یکدیگر حمله نکنند را توسط الگوریتم کم­ترین تناقض (Min Conflicts Algorithm) حل کرده­ایم. همچنین، این مسئله با فرض 16 در 16 بودن ابعاد صفحه و تعداد 16 وزیر نیز حل شده است.

تعریف مسئله: در این مسئله، هر وزیر را در یک ستون جدا در نظر گرفته­ایم و شماره ردیف هر وزیر را یک متغییر (variable) در نظر می­گیریم. بنابراین این مسئله دارای هشت متغییر است (برای حالت n=16 16 متغییر داریم) که هر کدام می­توانند 8 مقدار مختلف (برای حالت n=16 16 مقدار مختلف) اختیار کنند. که مقدار نسبت داده شده به یک متغییر، همان شماره ردیف وزیر مربوط به آن است. حال باید مقادیر این هشت متغییر را جوری انتخاب کنیم که هیچ دو وزیری نتوانند به یکدیگر حمله کنند.

در ادامه به شرح تک تک توابع پیاده­سازی می­پردازیم.

* **تابع Conflicts:** این تابع مشخص می­کند که به ازای یک مقدار (value) مشخص برای یک متغییر مشخص، چه تعداد نقض محدودیت (conflict) خواهیم داشت. اولین ورودی این تابع، متغییر مدنظر است که همان شماره ستون وزیر مربوطه می­باشد. دومین ورودی آن مقدار مدنظر برای متغییر است که همان شماره ردیف وزیر مربوطه می­باشد. و آخرین ورودی آن کل صفحه است. این تابع با پیمایش ردیف و قطر مربوط به متغییر ورودی، تعداد تناقضات آن را شمرده و برمی­گرداند.
* **تابع Initialize:** این تابع صفحه بازی را ایجاد می­کند و در هر ستون به صورت تصادفی یک وزیر قرار می­دهد. اگر خانه مقدار صفر داشته باشد یعنی خالی است و اگر مقدار آن یک باشد یعنی یک وزیر در آن قرار دارد.
* **تابع Solution:** این تابع صفحه بازی را دریافت کرده و مشخص می­کند که آیا حالت صفحه یک حالت هدف است یا خیر. این تابع یک مقدار True یا False بازمی­گرداند.
* **تابع ConflictedVariables:** این تابع تمام متغییرهایی که شرط محدودیت مسئله نقض می­کنند پیدا کرده و در یک لیست برمی­گرداند.
* **تابع MinConflicts:** این تابع الگوریتم Min Conflicts را پیاده­سازی می­کند و قسمت اصلی کد است. بدنه آن دقیقا مشابه شبه کد موجود در کتاب راسل پیاده­سازی شده است. با مطالعه شبه کد الگوریتم در شکل 1 و نگاه کردن به کد می­توان به راحتی کارکرد قسمت­های مختلف کد را متوجه شد.

****

شکل 1: شبه کد الگوریتم Min Conflicts

هر چقدر تعداد گام­های الگوریتم را افزایش دهیم، به دلیل وجود عناصر تصادفی زیاد در الگوریتم، ممکن است جوابی برای مسئله یافت نشود. از طرفی اجرای الگوریتم با گام­های زیاد، از نظر مدت زمان پردازش به صرفه نیست. بنابراین در بخش main، الگوریتم را آنقدر اجرا می­کنیم تا به جواب برسیم. طبق تجربه این الگوریتم پس چندین بار اجرا، حتما جواب را خواهد یافت.

لازم به ذکر است که ابعاد صفحه با متغییر n که در اول کد نوشته شده مشخص می­شود و با تغییر آن، ابعاد صفحه و تعداد وزیران تغییر خواهند کرد. بنابراین اگر جواب مسئله برای هشت وزیر مدنظر است، مقدار n را برابر با هشت قرار داده و اگر مسئله با 16 وزیر مدنظر است، مقدار n را برابر 16 قرار دهید.