

### ۳.

#### مساله ۸ وزیر با مانع امتیازی:

راه حل کلی این مساله کاملاً مشابه مساله دوم (بدون امتیاز) می‌باشد. تنها تفاوت آن وجود شروطی برای حرص شاخه‌هاست که باعث بهبود زمان اجرای الگوریتم می‌شود.

یک متغیر به نام score داریم که مقدار اولیه آن صفر است و در هر لحظه امتیاز حاصل از وزیرهایی که در جای خود قرار داده شده‌اند را نگهداری می‌کند. از آنجایی که امتیازها می‌توانند به صورت تصادفی توزیع شوند اطلاعی از امتیازی که قرار است از قرار دادن باقی وزیرها بدست آوریم نداریم، اما به طور قطع می‌دانیم امتیاز حاصل از این وزیرها از  $n * n$  ضربدر تعداد وزیرها کمتر و یا مساوی است (حالت تساوی وقتی پیش می‌آید که تنها یک وزیر باقی مانده باشد و بتوانیم آن وزیر را در خانه با امتیاز  $n * n$  قرار دهیم). بنابراین زمانی که وزیر را در خانه‌ای قرار می‌دهیم علاوه بر این که تهدید شدن آن را بررسی می‌کنیم باید بررسی کنیم که آیا با قرار دادن وزیر در این خانه شرط بالا برآورده می‌شود یا خیر. اگر شد می‌توانیم به ادامه الگوریتم پردازیم اما اگر نشد برای کسب امتیاز بیش از  $x$  وزیر به طور حتم نمی‌تواند در خانه مذکور قرار گیرد.

به بیان دقیق‌تر اگر  $p$  آرایه محل قرارگیری وزیرها بوده و  $s$  آرایه مربوط به ذخیره سازی امتیازها باشد، برای وزیر  $i$  ام باید داشته باشیم:

$$\text{score} + s[p[i]] + (n - i - 1) * (n * n) \geq x$$

که در آن امتیاز بدست آمده تا به این‌جا در score ذخیره می‌شود و  $s[p[i]]$  امتیاز بدست آمده از قرار دهی وزیر جدید می‌باشد. همچنین  $(n - i - 1)$  تعداد وزیرهای باقی‌مانده و  $n * n$  حداکثر امتیاز قابل کسب از هر وزیر می‌باشد.

در نهایت نیز پاسخ بدست آمده در صورتی معتبر است که امتیاز حاصل از آن بیش از  $x$  باشد.

#### تحلیل زمانی الگوریتم:

از آنجایی که در روش شاخه و حرص تضمینی بر حرص شاخه‌ها وجود ندارد در بدترین حالت این الگوریتم نیز مانند الگوریتم مساله قبل زمانی نمایی دارد.

## نمونه ورودی و خروجی برنامه:

```
Size of the board: 4
Minimum score(x): 0
Number of obstacles: 1
Position of the obstacle number 1: 1 2
Scores distribution:
1 2 3 4
5 6 7 8
9 10 11 12
13 14 15 16
```

```
-----
👑 ■ 👑 □
□ □ □ □
□ □ □ 👑
□ 👑 □ □
Score: 30
```

```
-----
□ ■ 👑 □
👑 □ □ □
□ □ □ 👑
□ 👑 □ □
Score: 34
```

Total number of answers: 2

شکل روبه‌رو نمونه ورودی و خروجی برنامه می‌باشد.

در این حالت مقدار  $x$  برابر با صفر در نظر گرفته

شده تا تمامی پاسخ‌های ممکن چاپ شوند.

```
Size of the board: 4
Minimum score(x): 31
Number of obstacles: 1
Position of the obstacle number 1: 1 2
Scores distribution:
1 2 3 4
5 6 7 8
9 10 11 12
13 14 15 16
```

```
-----
□ ■ 👑 □
👑 □ □ □
□ □ □ 👑
□ 👑 □ □
Score: 34
```

Total number of answers: 1

این بار مقدار  $x$  را برابر با ۳۱ در نظر می‌گیریم تا

تنها پاسخ دوم چاپ شود.