

```

for i in Range(N):
    Count = 0;
    A[i][0] = 1;
    for j in Range(N):
        if safe(1,1) == 1:
            Count += 1;
        A[j][1] = 1;
    Break

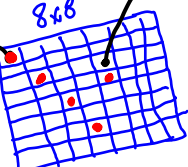
```



ستون ردیف  
A[0][0]

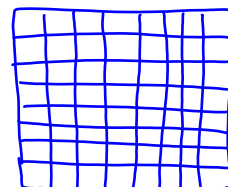
A<sub>0</sub>  
A<sub>1</sub>

A<sub>7</sub>



A[2][7]

A[0..7][0..7]



تمرین کامپیوتری شماره ۱ معماری کامپیوتر

نیمسال اول ۱۴۰۳-۱۴۰۴

زمان آپلود: روز شنبه ۵ آبان ۱۴۰۳ (ساعت ۲۳:۵۹)

در این تمرین کامپیوتری از شما خواسته شده است یک سخت‌افزار برای پیاده‌سازی الگوریتم هشت وزیر (8-Queen) در شطرنج پیدا کنید. شما باید ۸ وزیر را به گونه‌ای در صفحه‌ی شطرنج قرار دهید که هیچ دو وزیری یکدیگر را تهدید نکنند. کد پایتون زیر یک پیاده‌سازی از این الگوریتم را نشان می‌دهد که با استفاده از روش عقب‌گرد (Backtracking) پیاده‌سازی شده است.

```
def solveNQueens(board, col):
```

```
    if col == N:
```

```
        print(board)
```

```
        return True
```

```
    for i in range(N):
```

```
        if isSafe(board, i, col):
```

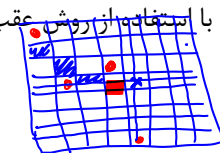
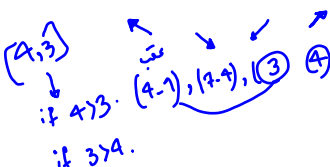
```
            board[i][col] = 1
```

```
            if solveNQueens(board, col + 1):
```

```
                return True
```

```
            board[i][col] = 0
```

```
    return False
```



q=i, l=col

isSafe(board, i, col) {

for m in range(N):

if board[i][m] == 1:

for n in range(N): return False;

if board[n][col] == 1: return False;

for i in range(4)

q=i, l=col;

while (0 < q < N) & (0 < l < N):

if board[q][l] == 1:

return False;

if i == 0:

return True;

q++, l++;

if i == 3:

return True;

if i == 2:

return True;

if i == 1:

return True;

if i == 0:

return True;

if i == 3:

return True;

if i == 2:

return True;

if i == 1:

return True;

if i == 0:

return True;

if i == 3:

return True;

if i == 2:

return True;

if i == 1:

return True;

if i == 0:

return True;

if i == 3:

return True;

if i == 2:

return True;

if i == 1:

return True;

if i == 0:

return True;

if i == 3:

return True;

- الگوریتم isSafe یک وزیر در ستون اول شروع می‌شود، سپس به ستون بعدی می‌رود و یک وزیر را در اولین ردیف امن آن ستون قرار می‌دهد.
- اگر الگوریتم به ستون ۸ برسد و تمام وزیرها در یک موقعیت امن قرار گیرند، الگوریتم خاتمه می‌یابد.
- اگر الگوریتم نتواند یک وزیر را در یک موقعیت امن در یک ستون خاص قرار دهد، به ستون قبلی برگشته و ردیف دیگری را امتحان می‌کند.
- تابع isSafe بررسی می‌کند که آیا قرار دادن یک وزیر روی یک سطر و ستون خاص ایمن است یا خیر.

راهنمایی:

- برای نگهداری محل وزیرها در صفحه‌ی شطرنج از ۸ رجیستر ۸ بیتی استفاده کنید.
- برای نگهداری محل قرارگیری آخرین وزیر از یک استک استفاده کنید تا امکان بازگرداندن حرکت آخر را فراهم کنید (عقب‌گرد).
- در صورت نیاز، برای شمارنده‌های سطر و ستون می‌توانید از شمارنده‌ها استفاده کنید.
- با فرمان شروع، صفحه‌ی شطرنج پاک شده و الگوریتم بالا اجرا می‌شود.
- در صورت پیدا کردن پاسخ، سیگنال خروجی Done به مدت یکی سیکل ساعت فعال می‌شود و در ۸ سیکل ساعت بعدی، محتوای صفحه‌ی شطرنج (یک سطر در هر مرحله) بر روی خروجی قرار می‌گیرد.
- در صورت پیدا نشدن پاسخ، سیگنال خروجی No\_Answer فعال می‌شود.

روش ارزیابی:

- پیاده‌سازی واحد مرتب‌سازی حبابی صد (۱۰۰) نمره دارد:

- ۲۵ نمره طراحی مسیر داده و واحد کنترل (تصویر مسیر داده و واحد کنترل باید آپلود شود)
- ۲۵ نمره روش کدینگ (مسیر داده به صورت ساختاری و واحد کنترل به روش هافمن)
- ۳۰ نمره برای صحت طراحی با داده‌های آزمون دانشجو (اسکرین شات از شکل موج خروجی شبیه‌سازی باید آپلود شود)
- ۲۰ نمره صحت طراحی با داده‌های آزمون توسط دستیاران آموزشی