

گزارش پروژه اول

(هشت وزیر)

مدل سازی مسئله

ابتدا این مسئله را به یک مسئله جستجو مدل کردم به این صورت که state ها را در آن مجموعه ای از زوج‌هایی در نظر گرفتم که هر زوج موقعیت یک وزیر را در صفحه شطرنج نشان می‌دهد. یعنی در واقع هر state، ۸ زوج است که نشان دهنده موقعیت‌های وزیرها داخل صفحه است.

استیت هدف یا goal این مسئله در واقع همان تعریف مسئله ۸ وزیر است یعنی state ای goal ماست که در آن هیچ دو وزیرای یکدیگر را تهدید نکنند. پس در این مسئله ما تنها یک هدف نداریم بلکه تعداد زیادی حالت هدف داریم که در همه آنها این شرط برقرار است و این موضوع را با استفاده از یک تابع چک می‌کنم.

در هر مرحله حرکات یکی به چپ، راست، بالا، پایین، راست بالا، راست پایین، چپ بالا و چپ پایین قابل انجام است پس در گراف فضای حالت ساخته شده از state هایی که در بالا توضیح داده شده‌است؛ هر state با state هایی که از طریق حرکات ذکر شده به آنها می‌توان رسید یعنی در صورت امکان (خالی بودن و داخل برد بودن) با ۶۴ نود همسایه است.

الگوریتم Breath First Search

ابتدا یک node ابتدایی داریم که همان حالت اولیه‌ی صفحه شطرنج است. در هر مرحله از بین node های داخل frontier مان اولین عنصر داخل صف را که در واقع کمترین عمق را دارد برمی‌داریم، چک می‌کنیم که این حالت را visit نکرده باشیم چرا که در آن صورت چون طبق bfs قبلاً آن حالت را با عمق کمتر مساوی ملاقات کرده‌ایم اگر به جواب برسد، از طریق همان حالت می‌رسد و ملاقات دوباره راس با عمق‌ای بیشتر چیزی به جواب‌های ما اضافه نمی‌کند.

اگر راس تکراری نبود آنرا به راس‌های ملاقات شده اضافه می‌کنیم و سپس frontier مان را با آن راس expand می‌کنیم. یعنی در واقع تمامی حرکات ممکن را برای تمامی وزیرهای داخل صفحه انجام می‌دهیم و در صورتی که قبلاً ملاقات نشده بودند (به علت همان استدلال بالا) آنها را به مجموعه رئوس frontier مان اضافه می‌کنیم.

البته برای بهتر کردن زمان الگوریتم قبل از اضافه کردن state به frontier آنها را چک می‌کنیم تا در صورتی که جواب نهایی باشند زودتر متوجه شویم و بیهوده راس های بیشتری ملاقات نکنیم تا زمان الگوریتم بدتر بشود. همچنین برای بهبود، برای نگهداری صف راس ها از deque استفاده کردم.

خروجی الگوریتم برای تست new_tests/test_a.csv

```
sadaf@ssadaf-ux310uq:~/Documents/Term6/AI/A1$ test_a
test_a: command not found
sadaf@ssadaf-ux310uq:~/Documents/Term6/AI/A1$ python3 bfs.py
```

```
o x o o o o o
o o o x o o o
o o o o o x o
o o o o o x o
o o x o o o o
o x o o o o o
o o o o o x o
o o o x o o o
```

Elapsed Time: 0.15802099999999997

Number of steps: 17181

Solution depth 3

```
o x o o o o o
o o o x o o o
o o o o o x o
o o o o o o x
o o x o o o o
x o o o o o o
o o o o o x o
o o o o x o o
```

خروجی الگوریتم برای تست new_tests/test_b.csv

```
sadaf@ssadaf-ux310uq:~/Documents/Term6/AI/A1$ test_b
test_b: command not found
sadaf@ssadaf-ux310uq:~/Documents/Term6/AI/A1$ python3 bfs.py
```

```
o x o o o o o
o o o x o o o
o o o o o x o
o o o x o o o x
o o o o o o o
o x o o o o o
o o o o x o o
o o o o x o o
```

Elapsed Time: 2.8110800000000005

Number of steps: 269217

Solution depth 4

```
o x o o o o o
o o o x o o o
o o o o o x o
o o o o o o x
o o x o o o o
x o o o o o o
o o o o o x o
o o o o x o o
```

خروجی الگوریتم برای تست new_tests/test_b.csv

```
sadaf@ssadaf-ux310uq:~/Documents/Term6/AI/A1$ test_c
test_c: command not found
sadaf@ssadaf-ux310uq:~/Documents/Term6/AI/A1$ python3 bfs.py
o x o o o o o o
o o x o o o o o
o o o o o o x o
o o o o o o o x
o o x o o o o o
o o o x o o o o
o o o o o o x o
o o o o x o o o

Elapsed Time: 32.329488000000005
Number of steps: 2627520
Solution depth 5
o x o o o o o o
o o o x o o o o
o o o o o x o o
o o o o o o o x
o o x o o o o o
x o o o o o o o
o o o o o o x o
o o o o x o o o
```

این الگوریتم روی تست‌های قبلی در زمان معقولی جواب ندادند چرا که حالت نهایی در آنها در عمق‌های بسیار بیش‌تر بود که باعث می‌شد تعداد راس‌های مان بسیار زیاد شده و هم حجم و هم زمان زیادی برای بررسی آنها گرفته شود.

```
sadaf@ssadaf-ux310uq:~/Documents/Term6/AI/A1$ _in1
_in1: command not found
sadaf@ssadaf-ux310uq:~/Documents/Term6/AI/A1$ python3 bfs.py
o o x o o o o
o o x o o x o o
o o o o o o o o
o o o o o o o x
o o o o o o o o
o o o o o o o o
o o o o o o x o
x x o o o o o o
o o o o o o x x
```

Process Name	User	% CPU	ID	Memory	Disk read	Disk write
indicator-messages-servi	sadaf	0	1586	940.0 KiB	N/A	N/A
livepatch-notif	sadaf	0	2201	N/A	N/A	N/A
nautilus	sadaf	0	2716	20.7 MiB	95.1 MiB	
nautilus-desktop	sadaf	0	1580	13.0 MiB	6.8 MiB	40.0
plugin_host	sadaf	0	2239	12.0 MiB	8.3 MiB	
pulseaudio	sadaf	0	1377	3.6 MiB	72.0 KiB	8.0
python3	sadaf	24	3800	5.7 GiB	13.2 MiB	
(sd-pam)	sadaf	0	1171	2.6 MiB	N/A	
seahorse	sadaf	0	3625	12.4 MiB	2.8 MiB	
ssh-agent	sadaf	0	1324	320.0 KiB	N/A	
sublime_text	sadaf	0	2220	37.1 MiB	9.3 MiB	4.0
systemd	sadaf	0	1170	1.5 MiB	17.8 MiB	5.0
update-notifier	sadaf	0	2195	4.8 MiB	58.2 MiB	5.7
Xorg	sadaf	0	1190	23.4 MiB	1.1 MiB	72.0
zeitgeist-daemon	sadaf	0	1985	1.3 MiB	1.9 MiB	2.5
zeitgeist-datahub	sadaf	0	1976	2.7 MiB	5.2 MiB	
zeitgeist-fts	sadaf	0	1991	2.4 MiB	4.3 MiB	14.0

```
sadaf@ssadaf-ux310uq:~/Documents/Tern6/AI/A1$ _in2
_in2: command not found
sadaf@ssadaf-ux310uq:~/Documents/Tern6/AI/A1$ python3 bfs.py
0 0 0 0 0 x 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 x 0 0 0 0 x 0 0
0 0 0 0 0 0 0 x 0
0 0 0 0 0 0 0 0 0
0 x 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 x 0 0
0 0 0 0 0 0 x 0

sadaf@ssadaf-ux310uq:~/Documents/Tern6/AI/A1$ _in3
_in3: command not found
sadaf@ssadaf-ux310uq:~/Documents/Tern6/AI/A1$ python3 bfs.py
0 0 0 0 0 x 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 x 0 0 0 0 x 0 0
0 0 0 0 0 0 0 x 0
0 x 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 x 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 x 0 0 0
0 0 0 0 0 x 0 0
```

Processes		Resources		File Systems	
Process Name	User	% CPU	ID	Memory	Disk read to Disk wr
indicator-messages-ser	sadaf	0	1594	924.0 KiB	N/A
livepatch-notif	sadaf	0	2072	N/A	N/A
nautilus	sadaf	0	2078	20.0 MiB	47.7 MiB
nautilus-desktop	sadaf	0	1586	12.8 MiB	6.8 MiB
plugin_host	sadaf	0	1967	12.0 MiB	8.3 MiB
pulseaudio	sadaf	0	1382	3.2 MiB	88.0 KiB
python3	sadaf	25	2059	4.6 GiB	30.4 MiB
(sd-pam)	sadaf	0	1175	2.6 MiB	N/A
seahorse	sadaf	0	2210	12.4 MiB	2.8 MiB
ssh-agent	sadaf	0	1328	320.0 KiB	N/A
sublime_text	sadaf	0	1954	27.3 MiB	9.3 MiB
systemd	sadaf	0	1174	1.5 MiB	4.6 MiB
update-notifier	sadaf	0	2065	4.8 MiB	58.0 MiB
Xorg	sadaf	0	1194	22.8 MiB	1.1 MiB
zeitgeist-daemon	sadaf	0	2005	1.2 MiB	1.4 MiB
zeitgeist-datahub	sadaf	0	2013	2.4 MiB	3.6 MiB
zeitgeist-fts	sadaf	0	2011	2.4 MiB	3.3 MiB

Processes		Resources		File Systems	
Process Name	User	% CPU	ID	Memory	Disk read to Disk wr
ibus-x11	sadaf	0	1446	4.5 MiB	N/A
indicator-application-ser	sadaf	0	1752	1.3 MiB	316.0 KiB
indicator-messages-ser	sadaf	0	1775	940.0 KiB	N/A
livepatch-notif	sadaf	0	2440	N/A	N/A
nautilus-desktop	sadaf	0	1771	12.8 MiB	6.9 MiB
plugin_host	sadaf	0	2152	12.0 MiB	8.3 MiB
pulseaudio	sadaf	0	1425	3.2 MiB	72.0 KiB
python3	sadaf	25	2217	3.5 GiB	30.4 MiB
(sd-pam)	sadaf	0	1199	2.6 MiB	N/A
seahorse	sadaf	0	2242	12.4 MiB	2.8 MiB
ssh-agent	sadaf	0	1372	316.0 KiB	N/A
sublime_text	sadaf	0	2139	20.0 MiB	9.0 MiB
systemd	sadaf	0	1198	1.5 MiB	51.0 MiB
update-notifier	sadaf	0	2433	4.7 MiB	1.8 MiB
Xorg	sadaf	0	1238	22.7 MiB	1.1 MiB
zeitgeist-daemon	sadaf	0	2172	1.2 MiB	1.4 MiB
zeitgeist-datahub	sadaf	0	2163	2.5 MiB	4.4 MiB

الغوريتم Iterative Deepening Search

ابتدا یک node ابتدایی داریم که همان حالت اولیه‌ی صفحه شطرنج است. در هر مرحله تابع DFS مان را با یک limit صدا می‌کند و تابع DFS تا آن عمق جلو می‌رود و اگر به نتیجه نرسید برای همسایه‌هایش صدا می‌کند تا جایی که به یک حالت goal برسد یا تمامی نود های این عمق ملاقات شده و تابع اصلی با limit ای یکی بیشتر DFS را صدا کند.

برای بهبود این الگوریتم نود های visit شده را همراه با عمق شان نگه داشتیم و در هر بار visit کردن نود چک کردم آیا این حالت با عمق کمتر مساوی ملاقات شده است یا خیر. اگر بله پس نیازی به ملاقات دوباره اش نیست و اگر خیر آنرا دوباره visit می‌کنیم.

برای بهبود زمانی اطلاعات نود های visit شده را در یک دیکشنری یا map از state به عمق نگهداری کردم تا دسترسی به آن سریع تر باشد.

جواب الگوریتم بهینه است چرا که همیشه از عمق کم به زیاد می‌رویم پس امکان ندارد ما برای اولین بار به جوابی برسیم ولی جوابی با عمق کمتر وجود داشته باشد چرا که حتما قبلا راس های با عمق کمتر visit شدند تا به این راس با عمق بیشتر رسیدیم.

جواب الگوریتم بهینه است چرا که همیشه از عمق کم شروع میکنیم و مسیر های به آن طول را visit می‌کنیم. در نتیجه امکان ندارد ما برای اولین بار به جوابی برسیم ولی جوابی با عمق کمتر وجود داشته باشد چرا که حتما قبل از تمام راس و مسیرهای با عمق کمتر visit شدند تا به این limit با عمق بیشتر رسیدیم.

خروجی الگوریتم برای تست new_tests/test_a.csv

```
sadaf@ssadaf-ux310uq:~/Documents/Term6/AI/A1$ test_a
test_a: command not found
sadaf@ssadaf-ux310uq:~/Documents/Term6/AI/A1$ python3 IDS.py
o x o o o o o o
o o o x o o o o
o o o o o x o o
o o o o o o x o
o o x o o o o o
o x o o o o o o
o o o o o o x o
o o o x o o o o

Elapsed Time: 0.133028000000000004
Number of steps: 6998
Solution depth 3
o x o o o o o o
o o o x o o o o
o o o o o x o o
o o o o o o o x
o o x o o o o o
x o o o o o o o
o o o o o o x o
o o o o x o o o
```

خروجی الگوریتم برای تست new_tests/test_b.csv

```
sadaf@ssadaf-ux310uq:~/Documents/Term6/AI/A1$ test_b
test_b: command not found
sadaf@ssadaf-ux310uq:~/Documents/Term6/AI/A1$ python3 IDS.py
o x o o o o o o
o o o x o o o o
o o o o o x o o
o o o x o o o x
o o o o o o o o
o x o o o o o o
o o o o x o o o
o o o o x o o o

Elapsed Time: 2.244644
Number of steps: 77192
Solution depth 4
o x o o o o o o
o o o x o o o o
o o o o o x o o
o o o o o o o x
o o x o o o o o
x o o o o o o o
o o o o o o x o
o o o o x o o o
```

خروجی الگوریتم برای تست new_tests/test_c.csv

```
sadaf@ssadaf-ux310uq:~/Documents/Term6/AI/A1$ test_c
test_c: command not found
sadaf@ssadaf-ux310uq:~/Documents/Term6/AI/A1$ python3 IDS.py
o x o o o o o o
o o x o o o o o
o o o o o o x o
o o o o o o o x
o o x o o o o o
o o o x o o o o
o o o o o o x o
o o o o x o o o

Elapsed Time: 29.3979400000000002
Number of steps: 569996
Solution depth 5
o x o o o o o o
o o o x o o o o
o o o o o x o o
o o o o o o o x
o o x o o o o o
x o o o o o o o
o o o o o o x o
o o o o x o o o
```

این الگوریتم روی تست‌های قبلی در زمان معقولی جواب ندادند چرا که حالت نهایی در آنها در عمق‌های بسیار بیش‌تر بود که باعث می‌شد تعداد راس‌های مان بسیار زیاد شده چرا که هر دفعه ضرب در ۶۴ می‌شوند و هم حجم و هم زمان زیادی برای بررسی آنها گرفته شود.

```
sadaf@ssadaf-ux310uq:~/Documents/Term6/AI/A1$ _in1
_in1: command not found
sadaf@ssadaf-ux310uq:~/Documents/Term6/AI/A1$ python3 IDS.py
o o x o o o o
o o x o o o o
o o o o o o o
o o o o o o x
o o o o o o x
o o o o o o x
x x o o o o o
o o o o o o x

[]
```

Process Name	User	% CPU	ID	Memory	Disk read	Disk v
ibus-x11	sadaf	0	1446	4.5 MiB	N/A	
indicator-application-ser	sadaf	0	1752	1.3 MiB	316.0 KiB	
indicator-messages-ser	sadaf	0	1775	940.0 KiB	N/A	
livepatch-notif	sadaf	0	2440	N/A	N/A	
nautilus-desktop	sadaf	0	1771	12.8 MiB	6.9 MiB	28.
plugin_host	sadaf	0	2152	12.0 MiB	8.3 MiB	
pulseaudio	sadaf	0	1425	3.6 MiB	72.0 KiB	8.
python3	sadaf	24	2518	1.6 GiB	4.0 KiB	
(sd-pam)	sadaf	0	1199	2.6 MiB	N/A	
ssh-agent	sadaf	0	1372	316.0 KiB	N/A	
sublime_text	sadaf	0	2139	27.6 MiB	9.2 MiB	332.
systemd	sadaf	0	1198	1.5 MiB	56.0 MiB	88.
update-notifier	sadaf	0	2433	4.7 MiB	1.8 MiB	5.
Xorg	sadaf	0	1238	22.7 MiB	1.1 MiB	72.
zeitgeist-daemon	sadaf	0	2172	1.2 MiB	1.4 MiB	96.
zeitgeist-datahub	sadaf	0	2163	2.5 MiB	4.4 MiB	
zeitgeist-fts	sadaf	0	2178	2.4 MiB	3.4 MiB	696.

```
sadaf@ssadaf-ux310uq:~/Documents/Term6/AI/A1$ _in2
_in2: command not found
sadaf@ssadaf-ux310uq:~/Documents/Term6/AI/A1$ python3 IDS.py
o o o o x o o o
o o o o o o o o
o X o o o o x o
o o o o o o o x
o o o o o o o o
o X o o o o o o
X o o o x o o o
o o o o o x o o

sadaf@ssadaf-ux310uq:~/Documents/Term6/AI/A1$ _in3
_in3: command not found
sadaf@ssadaf-ux310uq:~/Documents/Term6/AI/A1$ python3 IDS.py
o o o o x o o o
o o o o o o o o
o X o o o o x o
o o o o o o o x
o X o o o o o o
X o o o x o o o
o o o o o x o o

sadaf@ssadaf-ux310uq:~/Documents/Term6/AI/A1$
```

Process Name	User	% CPU	ID	Memory	Disk read	Disk wr
ibus-x11	sadaf	0	1446	4.5 MiB	N/A	
indicator-application-ser	sadaf	0	1752	1.3 MiB	316.0 KiB	
indicator-messages-servi	sadaf	0	1775	940.0 KiB	N/A	
livepatch-notif	sadaf	0	2440	N/A	N/A	
nautilus-desktop	sadaf	0	1771	12.8 MiB	6.9 MiB	28.0
plugin_host	sadaf	0	2152	12.0 MiB	8.3 MiB	
pulseaudio	sadaf	0	1425	3.6 MiB	72.0 KiB	8.0
python3	sadaf	25	2568	466.8 MiB	N/A	
(sd-pam)	sadaf	0	1199	2.6 MiB	N/A	
ssh-agent	sadaf	0	1372	316.0 KiB	N/A	
sublime_text	sadaf	0	2139	27.6 MiB	9.2 MiB	460.0
systemd	sadaf	0	1198	1.5 MiB	56.0 MiB	836.0
update-notifier	sadaf	0	2433	4.7 MiB	1.8 MiB	5.7
Xorg	sadaf	1	1238	22.7 MiB	1.1 MiB	72.0
zeitgeist-daemon	sadaf	0	2172	1.2 MiB	1.4 MiB	116.0
zeitgeist-datahub	sadaf	0	2163	2.5 MiB	4.4 MiB	
zeitgeist-fts	sadaf	0	2178	2.4 MiB	3.4 MiB	696.0

الگوریتم A^*

ابتدا یک node ابتدایی داریم که همان حالت اولیه‌ی صفحه شطرنج است. در هر مرحله مجموعه frontier مان را برحسب عددی که به آن نسبت می‌دهیم sort می‌کنیم و نود با کمترین عدد نسبت داده شده را در نظر می‌گیریم.

$$f(n) = g(n) + h(n)$$

عددی که به هر نود نسبت می‌دهیم مجموع هزینه تا به حال یعنی عمق نود و پیش بینی ما تا رسیدن به جواب یعنی heuristic مان می‌باشد.

heuristic را تعداد وزیرهایی در نظر می‌گیریم که در برد در حال تهدید شدن هستند. این heuristic یک admissible heuristic است چرا که همیشه در بهترین حالت برای اینکه وزیر ای از حالت تهدید در بیاید به یک حرکت نیاز داریم. پس حداقل به تعداد وزیر های در حال تهدید باید حرکت داشته باشیم تا به حالت goal برسیم.

پس به علت اینکه admissible heuristic داریم می‌دانیم که سرچ ما optimal search می‌شود.

خروجی الگوریتم برای تست new_tests/test_a.csv

```
sadaf@ssadaf-ux310uq:~/Documents/Term6/AI/A1$ test_a
test_a: command not found
sadaf@ssadaf-ux310uq:~/Documents/Term6/AI/A1$ python3 aStar.py
Elapsed Time: 0.06334899999999999
Number of steps: 35
Solution depth 5
o x o o o o o o
o o o x o o o o
o o o o o x o o
o o o o o o o x
o o x o o o o o
x o o o o o o o
o o o o o o x o
o o o o x o o o
```

خروجی الگوریتم برای تست new_tests/test_c.csv

```
sadaf@ssadaf-ux310uq:~/Documents/Term6/AI/A1$ test_c
test_c: command not found
sadaf@ssadaf-ux310uq:~/Documents/Term6/AI/A1$ python3 aStar.py
Elapsed Time: 0.063132000000000008
Number of steps: 35
Solution depth 5
o x o o o o o o
o o o x o o o o
o o o o o x o o
o o o o o o o x
o o x o o o o o
x o o o o o o o
o o o o o x o o
o o o o x o o o
```

تفاوت‌ها و مزیت‌ها

- الگوریتم BFS حافظه خیلی زیادی مصرف می‌کند اما حافظه مصرفی در Iterative deepening search و A* خیلی کمتر است.
- زمان الگوریتم A* بهتر سپس BFS و در نهایت Iterative deepening search است.
- تعداد حرکات تا رسیدن به جواب در A* کمتر سپس Iterative deepening search و در نهایت BFS است.