

## گزارش پروژه

فرمولاسیون(برای هر سه قسمت):

برنامه شامل یک کلاس کارت (شامل رنگ و عدد) و یک کلاس تخته بازی(board) است. برای حرکت دادن کارت ها از تابع move که دو عدد ردیف مبدا و مقصد را می گیرد و کارت آخر مبدا را به مقصد می برد استفاده می کنیم.

تابع goal\_test هم خود شامل دو تابع برای تست رنگ و عدد است و که برای هر ردیف فراخوانی می شوند.

تابعی به نام board\_move هم وجود دارد که حرکات مجاز تابع دریافتی را بر میگرداند

در ابتدا ما ورودی را گرفته و آن را تبدیل به یک آرایه دو بعدی از کارت ها می کنیم و سپس با آن یک تخته می سازیم

همچنین در تمامی الگوریتم ها ساخت نود جدید به این صورت است که ابتدا یک board جدید از روی parent board می سازیم(deep copy) و سپس حرکت مورد نظر را روی آن اعمال میکنیم . هر بورد نود والد خود و حرکتی که باعث تمایز خود و والدش شده را دارد.

: BFS

ابتدا تخته ی بازی اولیه را به تابع bfs می دهیم و تابع بعد از چک کردن goal نبودن در یک حلقه ی بینهایت اولین نود frontier را برداشته و explored اضافه می کند . سپس نود گسترش یافته و بعد از اینکه مطمئن شدیم این نود قبلا ساخته یا جستجو نشده مشخصات دیگرش را با آن اضافه می کنیم .سپس آن را برای هدف بود تست کرده و اگر هدف نبود به ته frontier اضافه می کنیم.

: IDS

این تابع بعد از فراخوانی در یک حلقه بی نهایت شروع به فراخوانی تابع dls برای limit های مختلف می کند

تابع dls نیز بعد از تست هدف روی ورودی به طور بازگشتی شروع به فراخوانی خودش میکند برای فرزندش میکند.وقتی یک نود به عمق cutoff می رسد روند برعکس شده و به عقب بر می گردیم و آخرین فرزند بعدی را گسترش می دهیم.

:A\*

تابع A\* کاملا مشابه bfs می باشد و تنها فرق آن این است که بعد از تست هدف هر نود, آن را به ته frontier نمی فرستیم بلکه frontier بر اساس f سورت شده است

$$F(n) = G(n) + H(n)$$

:Heuristic

هیوریستیک استفاده شده تعداد ردیف های نامرتب است. می دانیم با هر حرکت نهایتا یک ردیف مرتب خواهد شد بنابراین تعداد حرکات لازم برای سورت کامل قطعا بزرگتر یا مساوی تعداد ردیف های نامرتب خواهد بود

مقایسه:

تعداد گره های ساخته شده و بسط داده شده در IDS از همه بیشتر است.

تعداد گره های بسط داده شده در bfs کمتر از ids است و تعداد گره های ساخته شده چندین برابر بسط داده شده هاست

A\* بهترین عملکرد را داشته و کمترین نود ها را بسط می دهد و تعداد نود ساخته شده نیز با وجود اینکه چندین برابر بسط داده شده هاست باز هم بسیار کمتر از بقیه است.

همچنین عمق جواب در همه تقریبا یکسان است.