**تمرین 2 – مازندرانیان – 830402066**



کد از الگوریتم A\* برای حل پازل 8 استفاده می‌کند. ابتدا تابع get\_successors تمام حالات ممکن بعد از جابجایی خانه خالی را تولید می‌کند. تابع manhattan\_distance برای هر حالت مقدار تابع هزینه تخمینی را محاسبه می‌کند. تابع print\_state در هر مرحله وضعیت کنونی را نمایش می‌دهد. الگوریتم با استفاده از یک صف اولویت‌دار (heapq) اجرا می‌شود و در هر مرحله گره‌ای با کمترین هزینه را انتخاب می‌کند. اگر گره انتخابی همان وضعیت هدف باشد، مسیر حل نمایش داده می‌شود. در غیر این صورت، وضعیت فعلی به مجموعه بازدیدشده‌ها اضافه شده و همسایه‌های آن بررسی می‌شوند. برای هر همسایه، مقدار جدید g و h محاسبه شده و وضعیت در صف قرار می‌گیرد. در نهایت، تعداد حداقل حرکات موردنیاز برای رسیدن به وضعیت هدف چاپ می‌شود.



این کد از الگوریتم A\* برای حل پازل ۸ استفاده می‌کند و از یک معیار بهینه‌تر شامل مجموع فاصله مانهتن و تضادهای خطی استفاده می‌نماید. PuzzleNode هر وضعیت را همراه با والد، هزینه و مقدار تابع ارزیابی نگه می‌دارد. تابع get\_neighbors حرکت‌های ممکن را تولید می‌کند. تابع linear\_conflicts کاشی‌هایی را که مسیر یکدیگر را مسدود کرده‌اند شناسایی کرده و جریمه‌ای به تابع ارزیابی اضافه می‌کند. الگوریتم با استفاده از صف اولویت‌دار (open\_list) وضعیت‌های بهینه‌تر را زودتر بررسی کرده و از مجموعه closed\_set برای جلوگیری از تکرار وضعیت‌ها استفاده می‌کند. پس از رسیدن به حالت هدف، مسیر با استفاده از والدهای هر گره بازسازی شده و نمایش داده می‌شود.

