

گزارش تمرین اول

نگین اسماعیل زاده ۹۷۱۰۴۰۳۴

توابع :

در ابتدا از تابع `mapping` استفاده شده تا ورودی که به صورت یک فایل `xlm` است به یک لیست دو بعدی حاوی اطلاعات نقشه تبدیل شود. این تابع بدون ورودی فراخوانی میشود و فایل `SampleRoom.xlm` را میخواند و در خروجی یک لیست دو بعدی به نام `Map` میدهد.

تابع `A_star` بخش اصلی کد است که در ورودی یک لیست دو بعدی به اسم `visited_Map` را میگیرد (این لیست شامل اطلاعاتی است که تاکنون عامل از محیطش بدست آورده، خانه هایی از آن ها آگاهی موجود نیست `None` ذخیره می شوند) این تابع پس از یافتن مسیر بهینه ی خود تا باتری جهات حرکت به سمت آن را در یک لیست ذخیره میکند و این لیست را به عنوان ورودی به تابع `move` میدهد و آن را فراخوانی میکند.

تابع `move` یک لیست از جهات ها را به عنوان ورودی میگیرد و به ازای هر جهت امکان حرکت به آن جهت را بررسی میکند هر بررسی این تابع یک گام محسوب میشود و تصویر نقشه در هر گام نیز همزمان رسم میشود. اگر در مسیر حرکت عمل با دیوار مواجه شود تابع `move` ادامه حرکت را متوقف کرده و در همان مکان فعلی که حضور دارد مجدداً تابع `A_star` را فراخوانی میکند تا با اطلاعات جدید (`visited_Map` آپدیت شده در حرکات اخیر) مسیر دیگری را انتخاب کند همچنین شرط توقف مسئله که رسیدن با باتری هست در این تابع بررسی میشود.

تابع `graphical_view` تابعی است که جهت نمایش گرافیکی حرکت عامل در تابع `move` فراخوانی میشود در این تابع یک لیست دو بعدی (`Map`) به عنوان نقشه ورودی گرفته می شود و تصویر نقشه به شکل یک `heatmap` رسم میشود به این صورت که جایگاه منبع یا باتری با رنگ سبز ، جایگاه دیوار ها با رنگ آبی ، خانه های خالی با رنگ سفید و جایگاه فعلی عامل با رنگ سیاه نمایش داده می شود. ضمناً جهت تکرار متوالی تصاویر نقشه ها (در طول حرکت عامل) تصویر هر نقشه تنها به مدت ۱ ثانیه نمایش داده می شود.

مفروضات مسئله:

در این تمرین به طور کلی فرض شده است که عامل از محیط اطرافش در ابتدا چیزی نمی داند غیر از ۳ مورد : طول و عرض زمینی که در آن قرار گرفته است.

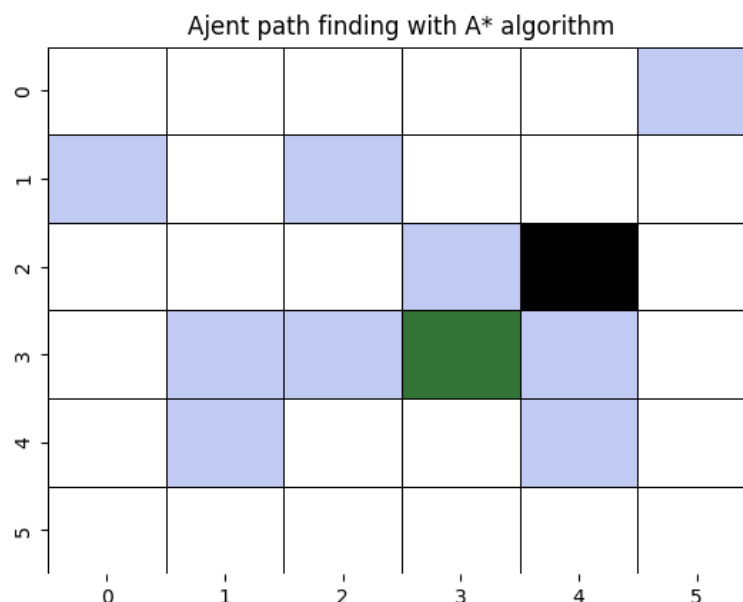
مکان فعلی خودش و بطور کلی هر مکانی که قبلا باز دید کرده (در آن ها قرار گرفته یا اگر دیوار بودند برخورد داشته). نتیجتا عامل دید از دور ندارد.

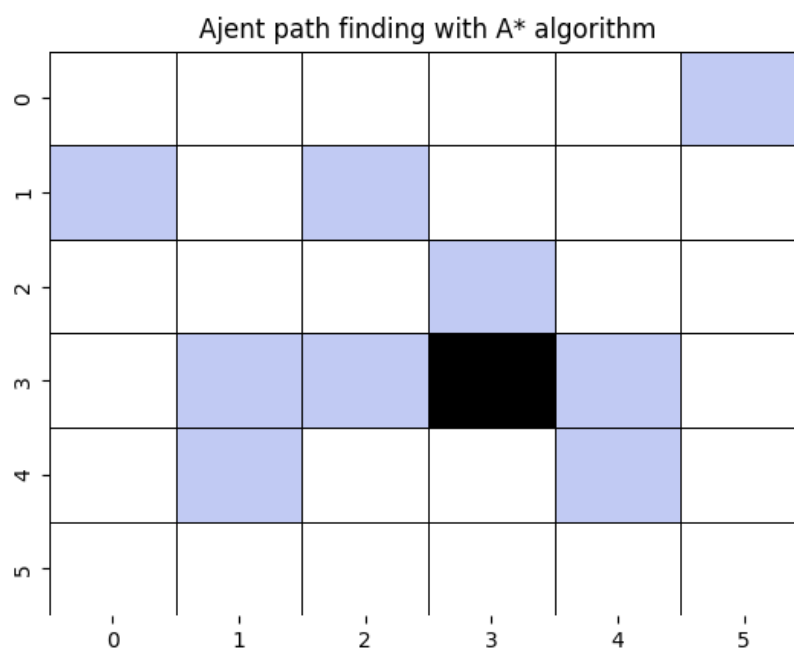
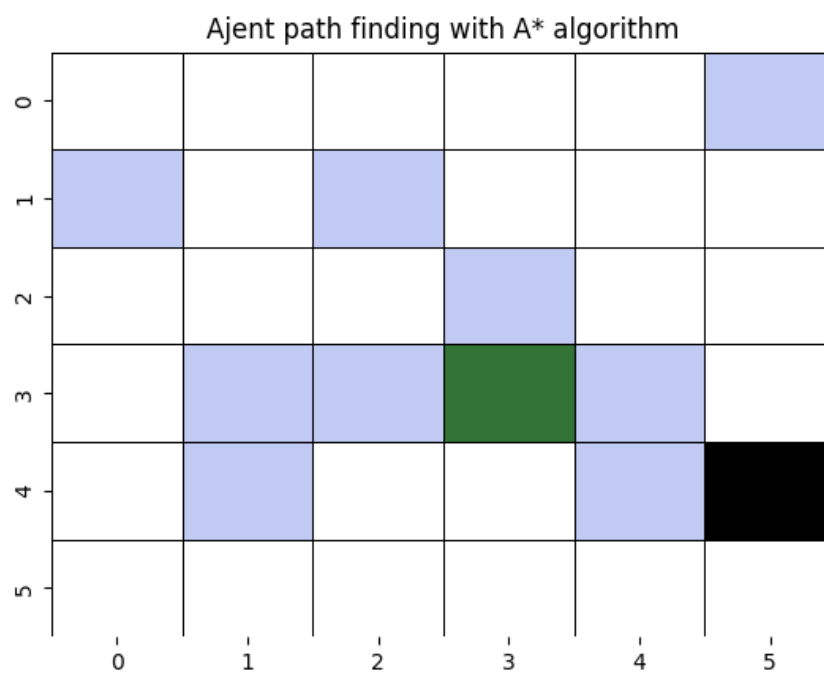
مختصات باتری (که البته سعی داشتیم در نسخه بعدی این مختصات را نیز حذف کنیم و یک مختصات تصادفی بین خانه های بازدید نشده جایگزین آن کنیم)

الگوریتم A^* :

در این تمرین برای خانه های مشاهده نشده تابع $g(n)$ برابر با فاصله منهتنی هر خانه از جایگاه فعلی ما و $h(n)$ فاصله منهتنی هر خانه از باتری فرض شده است. داریم $f(n) = g(n) + h(n)$ و بدیهی است که تابع f قابل قبول است زیرا همیشه مقدار آن از مقدار فاصله کمتر یا مساوی است (در صورت نبود مانع مساوی است).

نمونه خروجی این تمرین بر روی فایل SampleRoom در سه گام منتخب به صورت زیر است:





در این جا عامل توانست در مجموعا ۱۵ گام خود را به منبع برساند.

