

کلاس Node نشان دهنده یک خانه در صفحه است که اطلاعات موجود در آن شامل سطر و ستون آن خانه، و مسیر طی شده تا رسیدن به آن خانه است

در این کلاس چندین تابع نیز وجود دارد، تابع isValid بررسی می‌کند که آیا سطر و ستون این نود معتبر، یعنی بین ۱ تا ۴ هستند یا خیر، همینطور بررسی می‌کند که آیا نود مورد نظر، در مسیر رسیدن به خود دیده شده یا خیر، در این صورت معتبر نیست، زیرا در مسئله بیان شده هر خانه تنها یک بار میتواند دیده شود

تابع getNeighbors لیستی از فرزندان معتبر نود بر میگردد، برای ساخت این لیست، تابع ۸ فرزند میسازد، به این صورت که دو خانه به سمت پایین، بالا، چپ و راست حرکت کرده و سپس یک خانه میچرخد (نشان دهنده حرکت اسب)، سپس مسیر این فرزندان را برابر مسیر والد قرار میدهد، اگر فرزند معتبر بود (که در بالا تابع آن را بررسی کردیم) مسیر رسیدن به فرزند تکمیل شده (خود فرزند به آن اضافه میشود)، و سپس به لیست فرزندان اضافه شده، در نهایت این لیست خروجی تابع است

اپراتور \_\_eq\_\_ نیز برابر بودن سطر و ستون دو نود با یکدیگر را بررسی می‌کند

بخش جست و جوی برنامه در تابع bfs رخ میدهد، این تابع نود شروع و لیستی از سرباز ها را به عنوان ورودی گرفته، سپس یک صف ساخته، مسیر رسیدن به نود شروع را برابر خود نود قرار میدهد، سپس این نود را به صف اضافه کرده و جست و جو را شروع میکند

تا زمانی که صف خالی نشده، یک نود از صف میگیرد، لیست همسایه ها (فرزندان) آن نود را گرفته و به صف اضافه می‌کند، سپس لیست سرباز های خورده شده را برای هر فرزند بدست میآورد (برای این کار بین لیست خانه های سربازان و لیست مسیر رسیدن به نود فرزند اشتراک گرفته می‌شود)، سپس رکورد بهترین مسیر با بیشترین تعداد سربازان ذخیره میشود، اگر تعداد سربازان خورده شده برای این فرزند برابر تعداد کل سربازان بود، جست و جو متوقف شده، این مسیر به عنوان بهترین مسیر توسط تابع خروجی داده میشود

دقت کنید به دلیل چک شدن معتبر بودن همسایه ها در تابع getNeighbors، خانه ای در یک مسیر دوبار دیده نمی‌شود و همینطور از لوپ در جست و جو جلوگیری میشود

خروجی تابع یک سه تایی شامل موفق بودن یا نبودن در خوردن همه سرباز ها، بهترین مسیر پیدا شده و تعداد سربازان خورده شده در این مسیر است

در ادامه برنامه، ورودی های مسئله از کاربر دریافت شده، تابع bfs فراخوانی و خروجی آن به فرم خواسته شده در تمرین چاپ شده است