

## 1. فضاء الحالات (State Space)

فضاء الحالات يمثل كل المواقف أو الأوضاع الممكنة التي يمكن أن تصل إليها اللعبة بناءً على الحركات المتاحة. في هذه اللعبة، فضاء الحالات يتكون من شبكة  $n \times n$  (grid)، حيث يتم تمثيل كل حالة في الشبكة بمجموعات مختلفة من الخلايا التي تحتوي على قطع مختلفة.

### تمثيل الحالة:

- **خلايا حديدية:** هذه هي الخلايا التي تحتوي على قطع ثابتة لا يمكن تحريكها. يتم تمثيلها بلون "رمادي" في اللعبة.
- **خلايا مغناطيسية:** تحتوي على قطعتين مغناطيسيتين مختلفتين:
  - **المغناطيس الأحمر (Red Magnet):** يجذب إليه القطع الحديدية أو المغناطيسية الموجودة في نفس السطر أو العمود.
  - **المغناطيس الأرجواني (Purple Magnet):** يتنافر مع القطع الحديدية أو المغناطيسية المجاورة في نفس السطر أو العمود.
- **الخلايا الفارغة:** تمثل الأماكن التي يمكن للمغناطيسات أن تتحرك إليها. هذه الخلايا ليس بها أي قطعة حديدية أو مغناطيسية.
- **خلايا الأهداف البيضاء:** تمثل الخلايا التي يجب تغطيتها باستخدام القطع المغناطيسية أو الحديدية. تمثل أهداف اللعبة.

### محتويات الحالة:

- في كل حالة من الحالات، تحتوي الشبكة على معلومات حول مواقع القطع المغناطيسية والحديدية.
- كما تحتوي على حالة كل خلية: هل هي فارغة؟ تحتوي على قطعة مغناطيسية أو حديدية؟ أم هي خلية هدف؟

## 2. الحالة الابتدائية (Initial State)

في بداية اللعبة، يتم تحديد مواقع القطع المغناطيسية والحديدية في الشبكة. يتم تحديد هذه المواقع بناءً على التكوين الأولي للرقعة.

### تكوين الحالة الابتدائية:

- **المغناطيسات:** يتم تحديد موقع كل قطعة مغناطيسية (أحمر أو أرجواني) في الشبكة. هذه القطع تكون متحركة.
- **القطع الحديدية:** يتم تحديد مواقع القطع الحديدية في الشبكة. هذه القطع ثابتة، ولا تتحرك إلا عند تأثير المغناطيسات.
- **الخلايا البيضاء:** يتم تحديد الخلايا البيضاء في الشبكة التي يجب تغطيتها.

### 3.العمليات(Actions)

العمليات هي الحركات التي يمكن تنفيذها على الشبكة لتحريك القطع المغناطيسية وتحقيق الهدف.

#### تحريك القطع المغناطيسية:

- **Purple magnet () :** يمكن تحريكها إلى أي خلية فارغة في الشبكة. عند تحريكها، تتسبب في "التنافر" مع القطع الحديدية أو المغناطيسية المجاورة في نفس السطر أو العمود. بمعنى آخر، تتحرك القطع المجاورة بعيداً عنها في الاتجاه المعاكس لحركتها.

مثال :إذا كانت القطعة البنفسجية في الخلية (2, 2)، وكانت هناك قطعة حديدية في الخلية (2, 1)، فإن القطعة الحديدية ستتحرك في الاتجاه المعاكس (أي إلى الخلية (2, 3))

- **Red magnet () :** يمكن تحريكها إلى أي خلية فارغة أيضاً. عند تحريكها، تتسبب في "التجاذب" مع القطع الحديدية أو المغناطيسية المجاورة في نفس السطر أو العمود. بمعنى آخر، تتحرك القطع المجاورة نحوها.

مثال :إذا كانت القطعة الحمراء في الخلية (4, 0) وكانت هناك قطعة حديدية في الخلية (3, 0)، فإن القطعة الحديدية ستتحرك باتجاه القطعة الحمراء إلى الخلية (2, 0)

**move\_magnet :** تابع لفحص المكان الذي ساقوم بنقل المغنايس عليه ان كان فارغ او ابيض

**play console :** تابع لاختبار اللعب بالكونسول حيث اقوم بادخال الحركات يدوية على شكل احداثيات ثم يطبع لي الحالة بعد الحركة المدخلة

**play gui :** تابع للعب باستخدام واجهة رسومية

#### القيود على التحريك:

- **القطع الرمادية :** هذه القطع لا تتحرك مباشرة. يمكنها التحرك فقط إذا تأثرت بحركة قطعة مغناطيسية أحمر أو أرجواني
- **الحركات المحظورة :** لا يمكن تحريك القطع المغناطيسية أو الحديدية إلى خلايا غير فارغة أو خارج حدود الشبكة.

### 4.الحالات النهائية(Goal States)

الحالة النهائية تتحقق عندما يتم تغطية جميع الخلايا التي تحتوي على أهداف (الخلايا البيضاء) باستخدام القطع المغناطيسية أو الحديدية. يجب أن تتحقق هذه الحالة مع الالتزام بجميع قواعد اللعبة

## 5.الحل(Solution)

الحل هو مجموعة من الحركات التي يمكن تطبيقها على الرقعة للوصول من الحالة الابتدائية إلى الحالة النهائية. يتم ذلك عبر تطبيق العمليات المذكورة (تحريك القطع المغناطيسية) بشكل استراتيجي.

### البحث عن الحلول:

- **البحث بالعرض (BFS):** يمكن استخدام البحث بالعرض لاستكشاف الحلول بأقل عدد من الحركات. هذا يعني استكشاف جميع الحركات الممكنة في كل مستوى من الشجرة قبل الانتقال إلى المستوى التالي.

### البنى المستخدمة:

- **الطابور (Queue):** يتم تخزين الحالات في الطابور، مما يسمح بمعالجة الحالات في ترتيب وصولها، بحيث يتم معالجة الحالات الأقرب أولاً.
- **مجموعة الزيارات (Visited Set):** تُستخدم أيضاً هنا لتخزين الحالات التي تمت زيارتها.

تتم إضافة الحركات الممكنة إلى **الطابور** مع تحديث المسار path الذي يوضح الحركات التي تم تنفيذها للوصول إلى تلك الحالة.

عندما يتم العثور على الحل، يتم ببساطة إرجاع المسار دون أي طباعة إضافية، لكن لو كنت ترغب في تتبع الحركات مثل DFS، يمكنك إضافة print في المكان الذي يتم فيه العثور على الحل.

- **البحث بالعمق (DFS):** في البحث بالعمق، يتم استكشاف كل مسار ممكن حتى النهاية قبل الرجوع إلى الوراء وتجربة مسار آخر.

### البنى المستخدمة:

- **المكدس (Stack):** يتم تخزين الحالات في المكدس، مما يسمح بالعودة إلى الحالة السابقة بمجرد استنفاد الخيارات الحالية. هذه البنية تُستخدم لأن DFS يقوم بالاستكشاف بشكل "عميق".
- **مجموعة الزيارات (Visited Set):** تُستخدم لتخزين الحالات التي تمت زيارتها لتجنب التكرار وعدم الدخول في حلقات لانهاية.
- 
- تم فحص جميع الحركات الممكنة في كل خطوة (كما يظهر في الحلقات المزدوجة). إذا كانت الحركة ممكنة وتم اكتشاف حالة جديدة، تتم إضافتها إلى المكدس مع تتبع الحركات التي تم إجراؤها (new\_path).

- عندما يتم إيجاد الحل (أي عندما نجد حالة الهدف)، يتم طباعة المسار الذي تم اتباعه لتصل إلى الهدف باستخدام `print(f"Move magnet from ({move[0]}, {move[1]}) to ({move[2]}, {move[3]})").` هذا يوضح لك الحركات التي تمت خطوة بخطوة.