

# A-Star. Пятнашки

Гусев Илья, Булгаков Илья

Московский физико-технический институт

Москва, 2019

# Содержание

- 1 Правила игр Пятнашки и Восьминашки
- 2 Разбор игры Восьминашки
- 3 Разбор игры Пятнашки
- 4 Алгоритм  $A^*$
- 5 Эвристики для Пятнашек

# Игра Пятнашки



# Правила игры Пятнашки



- 15 карточек с числами от 1 до 15
- Одно пустое поле
- Доступна операция сдвига карточки с числом на соседнее поле, если оно пусто
- Цель игры — перемещая костяшки по коробке, добиться упорядочивания их по номерам, желательно сделав как можно меньше перемещений.

# Правила игры Восьминашки



- 8 карточек с числами от 1 до 8
- Одно пустое поле
- Доступна операция сдвига карточки с числом на соседнее поле, если оно пусто
- Цель игры — перемещая костяшки по коробке, добиться упорядочивания их по номерам, желательно сделав как можно меньше перемещений.

# Игра Восьминашки

Любую ли перестановку можно получить из другой?



# Игра Восьминашки

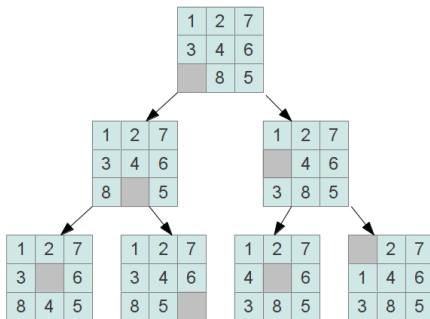
- Перестановка множества  $S$  содержит инверсию элементов  $i_j$  и  $i_k$ , если нарушен их естественный порядок расположения, т.е. больший элемент расположен левее меньшего:  $i_j > i_k$  при  $(j < k)$ .
- Считаем число инверсий в перестановке  
12345678 - 0 инверсий  
12345687 - 1 инверсия
- Меняет ли ход в игре восьминашки четность перестановки?



# Игра Восьминашки

Как будем искать решение задачи? Помогут графы!

- Состояние доски - вершина. Ход - ребро.
- Нужно найти путь из одной вершины в другую.
- Какие есть алгоритмы? Взвешенный у нас граф или нет?



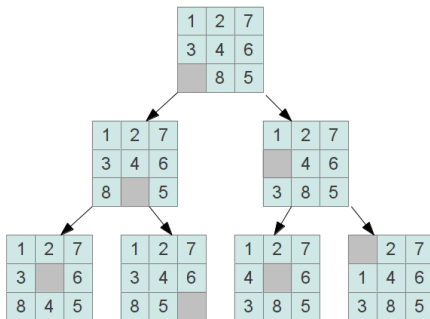
.....



# Игра Восьминашки

Почему поиск в ширину работает?

- Сколько у нас перестановок?  $8!/2 = \text{ок. } 180000$ . Обойти реально.
- Как будем представлять одно состояние доски? Какую структуру использовать?



# Игра Пятнашки

Как будем решать? Надо ли что-то менять по сравнению с восьминашками?



# Игра Пятнашки

- Меняется условие на достижимость одной перестановки из другой.
- Как будем представлять одно состояние доски? Какую структуру использовать?
- Обход в ширину не поможет  $15!/2 = \text{ок. } 600 \text{ млрд.}$  Обойти нереально.

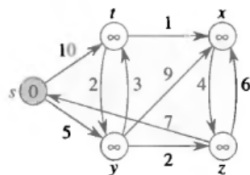


# Игра Пятнашки

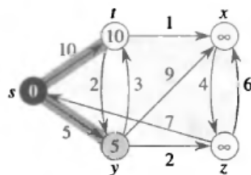
- Условие на достижимость: четность перестановки без 0 + номер строки, где находится 0.
- Представление состояния: плотная упаковка. 64-битное целое, на каждую ячейку по 4 бита.
- Вместо обхода в ширину – **другой алгоритм**, более умный!



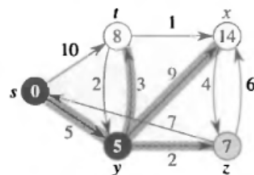
# Вспоминаем: алгоритм Дейкстры



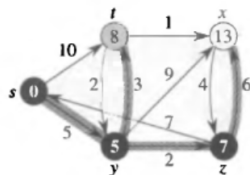
(a)



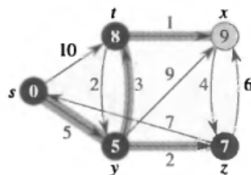
(б)



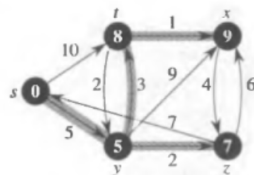
(v)



(r)

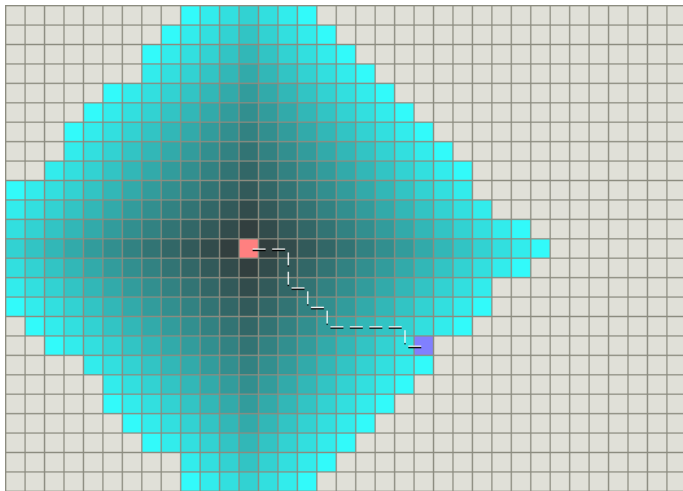


(д)



(e)

# Вспоминаем: алгоритм Дейкстры



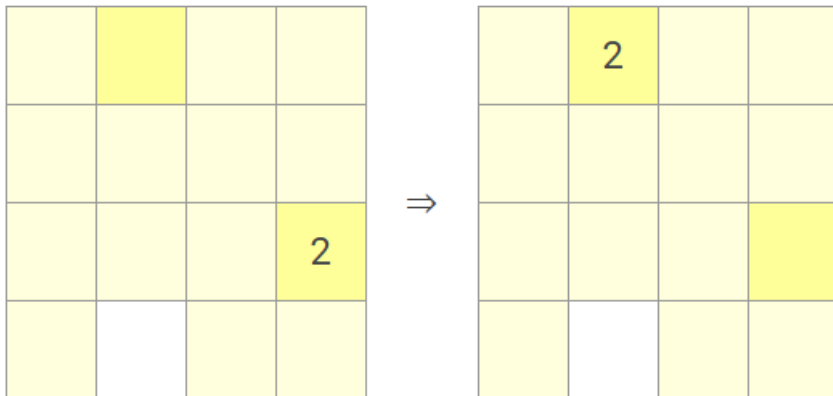
# Эвристика

Эвристика — оценка расстояние от вершины до целевой вершины.  
Мы берём эвристику, исходя из знаний о среде

# Эвристика: примеры

Среда: квадратная сетка с 4 направлениями движения

Эвристика: манхэттенское расстояние

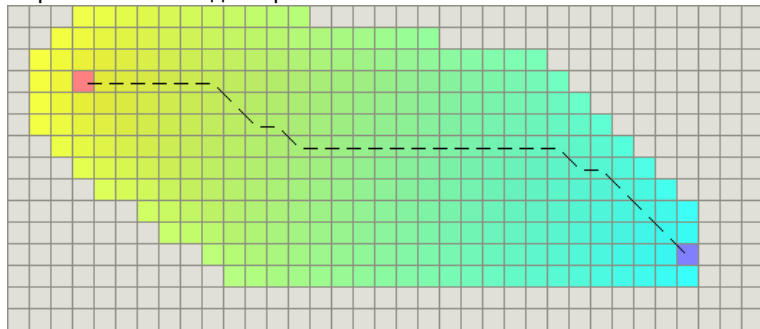




# Эвристика: примеры

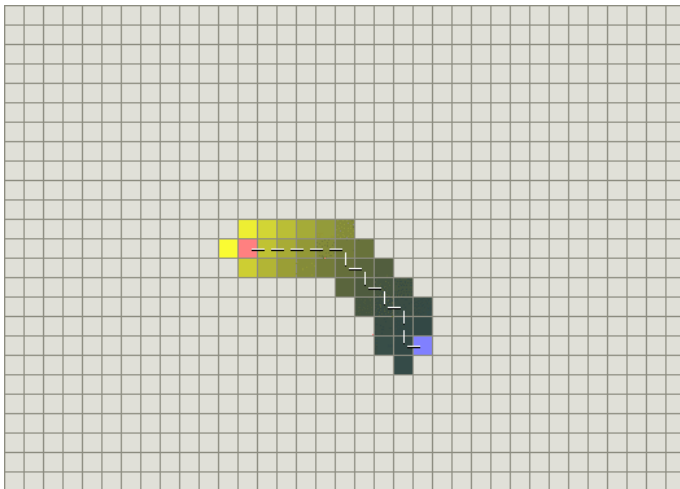
Среда: произвольная карта

Эвристика: евклидово расстояние

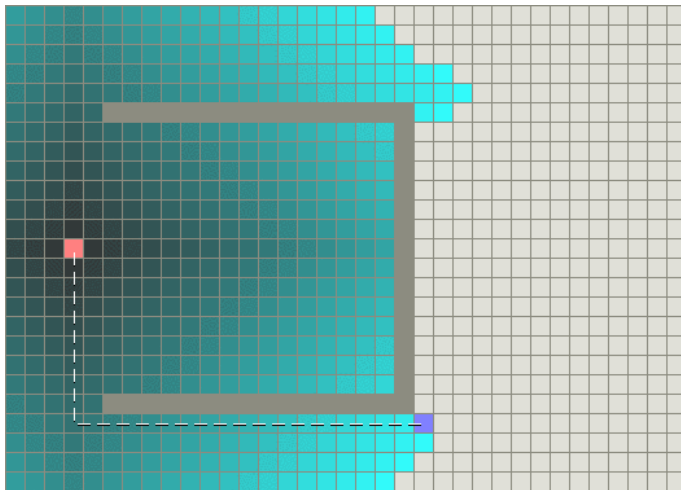


# Greedy Best-First-Search

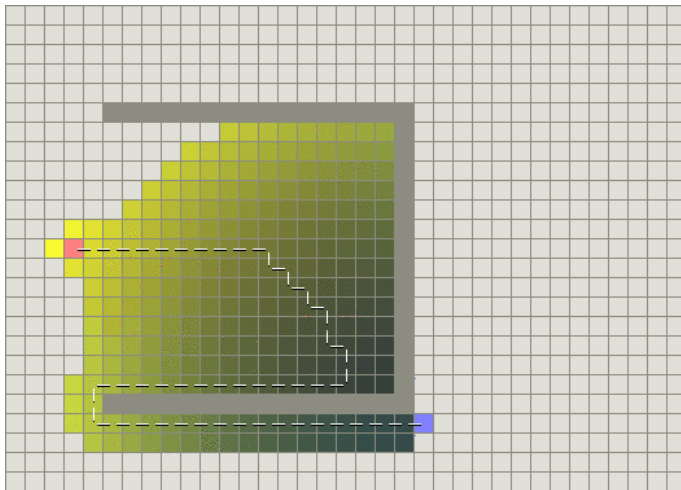
Минимизация расстояния от  $S \Rightarrow$  максимизация эвристики до  $T$



# Препятствия (Дейкстра)



# Препятствия (GBFS)



## A\*

A\* = Дейкстра + эвристика

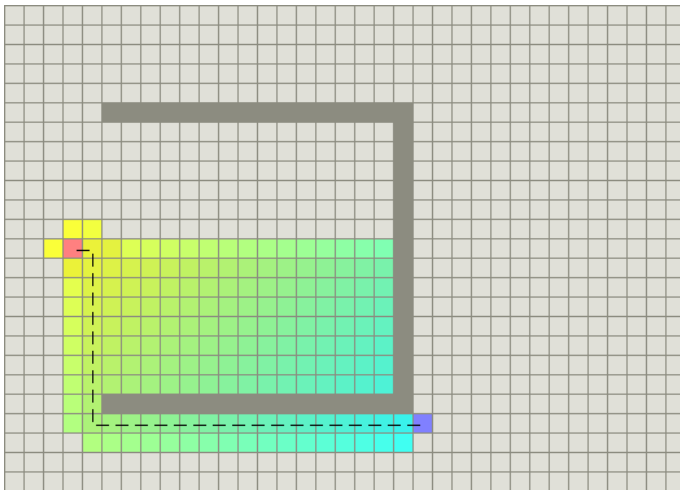
$f(n) = g(n) + h(n)$ , где

$g(n)$  - минимальное расстояние от стартовой точки (вершины)

$h(n)$  - эвристическое расстояние до конечной точки (вершины)

$f(n)$  - итоговая оценка точки (вершины)

# Препятствия (A\*)



# Свойства эвристики

- 1  $\forall n, h(n) = 0 \Rightarrow$  обычный Дейкстра с гарантией кратчайшего пути
- 2 Допустимость:  $\forall n, h(n) \leq f(n, T) \Rightarrow A^*$  с гарантией кратчайшего пути
- 3 Чем меньше  $h(n)$ , тем больше вершин обходится
- 4 Монотонность:  $\forall n, p : \exists E(n, p) \rightarrow h(n) - h(p) \leq w(n, p), h(T) = 0$
- 5 Монотонность  $\Rightarrow$  допустимость
- 6  $\forall n, h(n) = f(n, T) \Rightarrow$  всегда идеальный путь
- 7  $h(n)$  может переоценивать  $f(n, T)$ , но тогда путь может быть неоптимальным

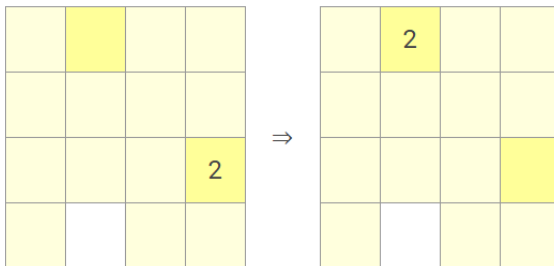
# Эвристики для Пятнашек

- 1 Манхэттенское расстояние
- 2 Линейный конфликт
- 3 Угловые фишки
- 4 Последний ход



# Эвристики для Пятнашек

**Манхэттенское расстояние** – это сумма расстояний по строкам и столбцам от текущего расположения костяшки до ее правильной позиции  
На примере ниже манхэттенское расстояние – 4.

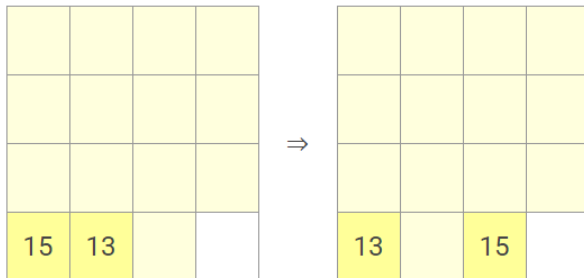


# Эвристики для Пятнашек

Считается, что костяшка I и костяшка J находятся в **линейном конфликте** по строке, если они обе стоят в своей строке, но костяшка I находится левее костяшки J, хотя на самом деле должна быть справа.

На этой схеме  $I = 15$ ,  $J = 13$ .

Мы должны подвинуть одну из костяшек со строки, поэтому можем добавить 2 к манхэттенскому расстоянию. Аналогичным образом рассматривается линейный конфликт по столбцу.



# Эвристики для Пятнашек

**Угловые фишки.** Рассмотрим правый верхний угол. Пусть «3» или «8» стоит на своей позиции, а на месте «4» — любая другая костяшка. В таком случае, чтобы поставить «4» на место, костяшки придется подвинуть. Для этого добавим 2 к манхэттенскому расстоянию. Если «3» или «8» участвует в линейном конфликте (linear conflict), то двойка уже добавлена.

Аналогично с левым верхним и левым нижним углами. Правый нижний угол в эвристике не рассматривается, так как очень сложно скомбинировать этот случай с эвристикой «Последний ход».

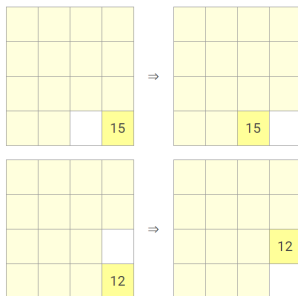
		3	!4

			!4
			8

# Эвристики для Пятнашек

**Последний ход.** На последнем ходу мы либо двигаем костяшку «15» влево, либо «12» — вверх

Если костяшки не находятся на требуемых позициях («15» — в крайнем правом ряду или «12» — в самой нижней строке), манхэттенское расстояние не учитывает переход через угол. Поэтому мы можем добавить к нему 2.



# Полезные ссылки I



[Amit's A\\* Pages From Red Blob Games](http://theory.stanford.edu/~amitp/GameProgramming/)

<http://theory.stanford.edu/~amitp/GameProgramming/>



[neerc.ifmo.ru: Алгоритм A\\*](https://bit.ly/2q3RbJg)

<https://bit.ly/2q3RbJg>



[Wiki: A\\* search algorithm](https://en.wikipedia.org/wiki/A*_search_algorithm)

[https://en.wikipedia.org/wiki/A\\*\\_search\\_algorithm](https://en.wikipedia.org/wiki/A*_search_algorithm)



[Wiki: admissible heuristic](https://en.wikipedia.org/wiki/Admissible_heuristic)

[https://en.wikipedia.org/wiki/Admissible\\_heuristic](https://en.wikipedia.org/wiki/Admissible_heuristic)