

## Алгоритм мого мегабота

Алгоритм дуже простий - вибирає координати, які мають найкоротшу векторну відстань якраз за її формулою дорівнює кореню квадратному із суми квадратів різниць відповідних координат.

Тут якраз вираховується ця відстань: point1 - мої координати, point2- координати ворога.

```
1 def distance(point1, point2):
2     '''
3     Returns vector distance between my coordinates and enemy's coordinates
4     '''
5     return sqrt((point2[0] - point1[0])**2 + (point2[1] - point1[1])**2)
6
```

Функція котра повертає координати з найкоротшою відстанню, якщо в мене чи в противника немає ходів, тоді повертає пустий список, щоб просто заповнити поле беручи перші можливі координати і кожен раз не вираховувати мінімальну відстань від пустих координат:

```
1 def shortest_distance(coords: tuple[list[tuple[int]]]):
2     '''
3     Returns closest coordinate to the opponent
4     '''
5     min_distance = float('inf')
6
7     my_move = coords[0]
8
9     enemy_move = coords[1]
10
11     if my_move and enemy_move:
12         for m_m in my_move:
13             for e_m in enemy_move:
14                 dist = distance(m_m, e_m)
15                 if dist < min_distance:
16                     min_distance = dist
17                     closest_coord = m_m
18
19     return closest_coord
20
21 return []
22
23
24
25
26
27
```

Алгоритм дуже простий і на великій карті показує себе найкраще.  
На середній багато залежить від розташування, тобто який ти плеєр.  
На малій ж карті показує себе максимально дивно, якщо ти перший  
гравець(зелений), тоді алгоритм максимально вразливий, коли ж другий  
плеєр(синій), то ніби все працює належним чином.