南方科技大学
SOUTHERN UNIVERSITY OF SCIENCE AND TECHNOLOGY

Lab 4 - Practice - BFS

CS208 Algorithm Design and Analysis

Yang Xu

xuyang@sustech.edu.cn

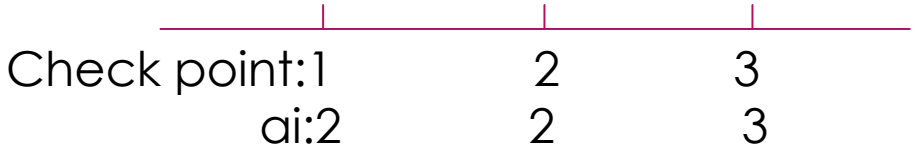
Slides adapted from Yao Zhao zhaoy6@sustech.edu.cn

Question 1:

- FluffyT, the super bunny, successfully escaped from Satori's bunny shop. She then ran into an alley with N check points.
- As FluffyT is a super bunny, she can build a portal between check point i and a_i using 1 second and travel through within no time. However, she cannot travel in the opposite direction (from a_i to i). Also, she can run from check point i to check point $i-1$ and $i+1$ using 1 second.
- FluffyT is currently at check point 1. Can you tell her the minimum time to get to each check point?

Sample Input 1

n = 3
i = 1, 2, 3
ai = 2, 2, 3



minimum time

Check point 1: 1->1 0

Check point 2:
1->2 1
1->a1=2 1

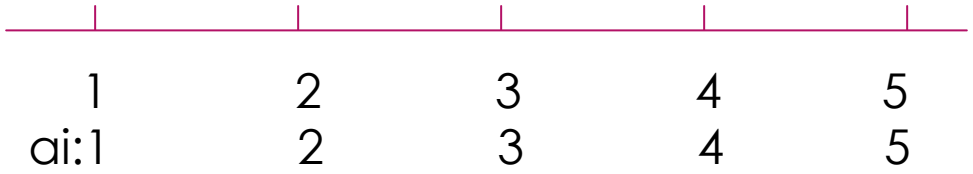
Check point 3:
1->2 ->3 2
1->a1=2->3 2

Sample Output
0 1 2

minimum time

Sample Input 2

n = 5
i = 1 2 3 4 5
ai= 1 2 3 4 5



Check point 1: 1 0

Check point 2: 1->2 1

Check point 3: 1->2 ->3 2

Check point 4: 1->2->3->4 3

Check point 5: 1->2->3->4->5 4

Sample Output 2
0 1 2 3 4

Sample Input 3

n =7
i = 1 2 3 4 5 6 7
ai = 4 4 4 4 7 7 7

minimum time

Check point 1: 1 0

Check point 2: 1->2 1

Check point 3: 1->2 ->3 2

Check point 4: 1->a1:4 1

Check point 5: 1-> a1:4 ->5 2

Check point 6: 1-> a1:4 ->5 ->6 3

Check point 7: 1-> a1:4 ->5 ->a5:7 3

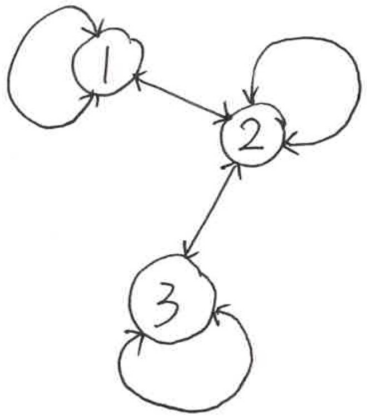
Hint: Try BFS on the graph

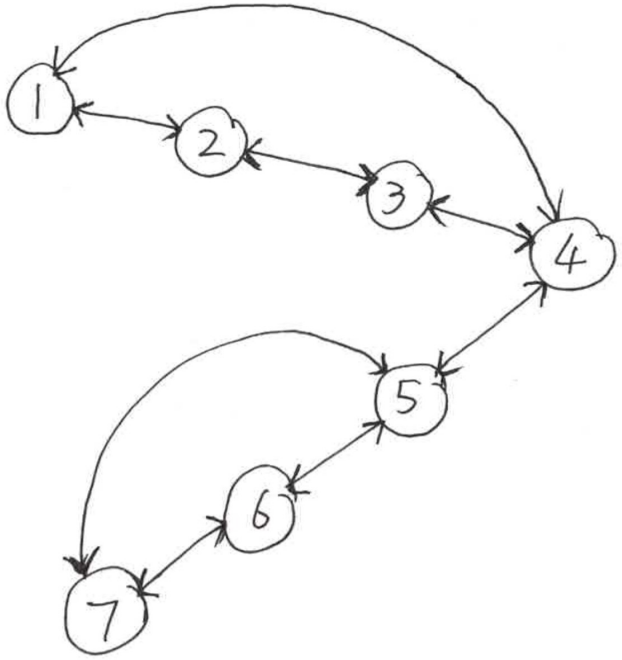
Sample Output 3
0 1 2 1 2 3 3

Analysis

Check point:1	2	3
ai:2	2	3

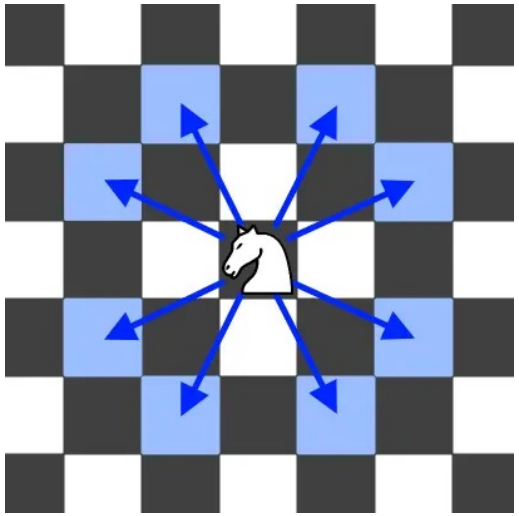
1	2	3	4	5	6	7
a i:4	4	4	4	7	7	7

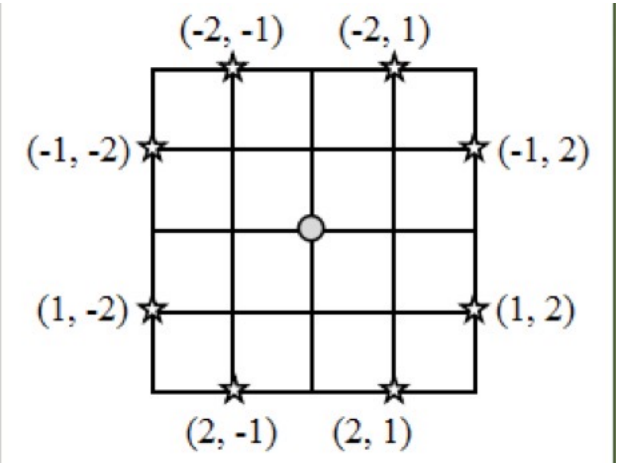




Question 2:

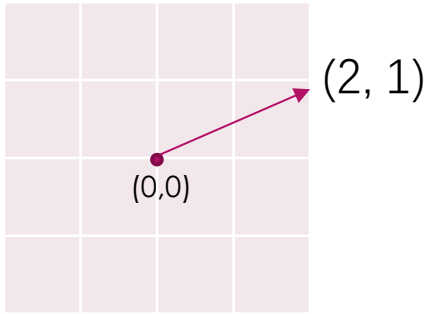
- A knight has 8 possible moves it can make, as illustrated below. Each move is two squares in a cardinal direction, then one square in an orthogonal direction.



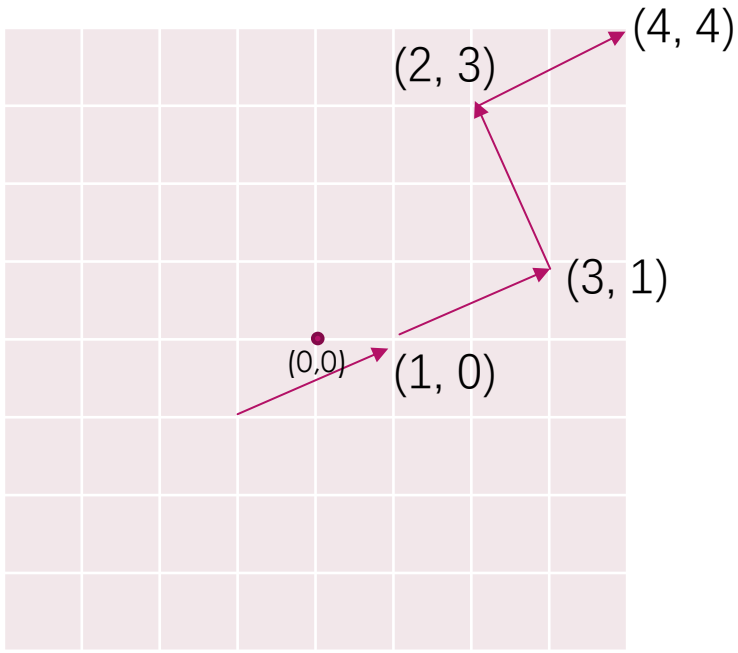


- On a n*n chess board, given the starting point (x1, y1) and the ending point (x2, y2), calculate the minimum number of moves a knight needs to make from the starting point to the ending point. If the knight cannot reach the ending point, return -1.

Example 1:
Input: n = 4, x1 = 0, y1 = 0, x2 = 2, y2 = 1
Output: 1
Explanation: [0, 0] → [2, 1]



Example 2:
Input: n = 8, x1 = -1, y1 = -1, x2 = 4, y2 = 4
Output: 4
Explanation: [-1, -1] → [1, 0] → [3, 1] → [2, 3] → [4, 4]

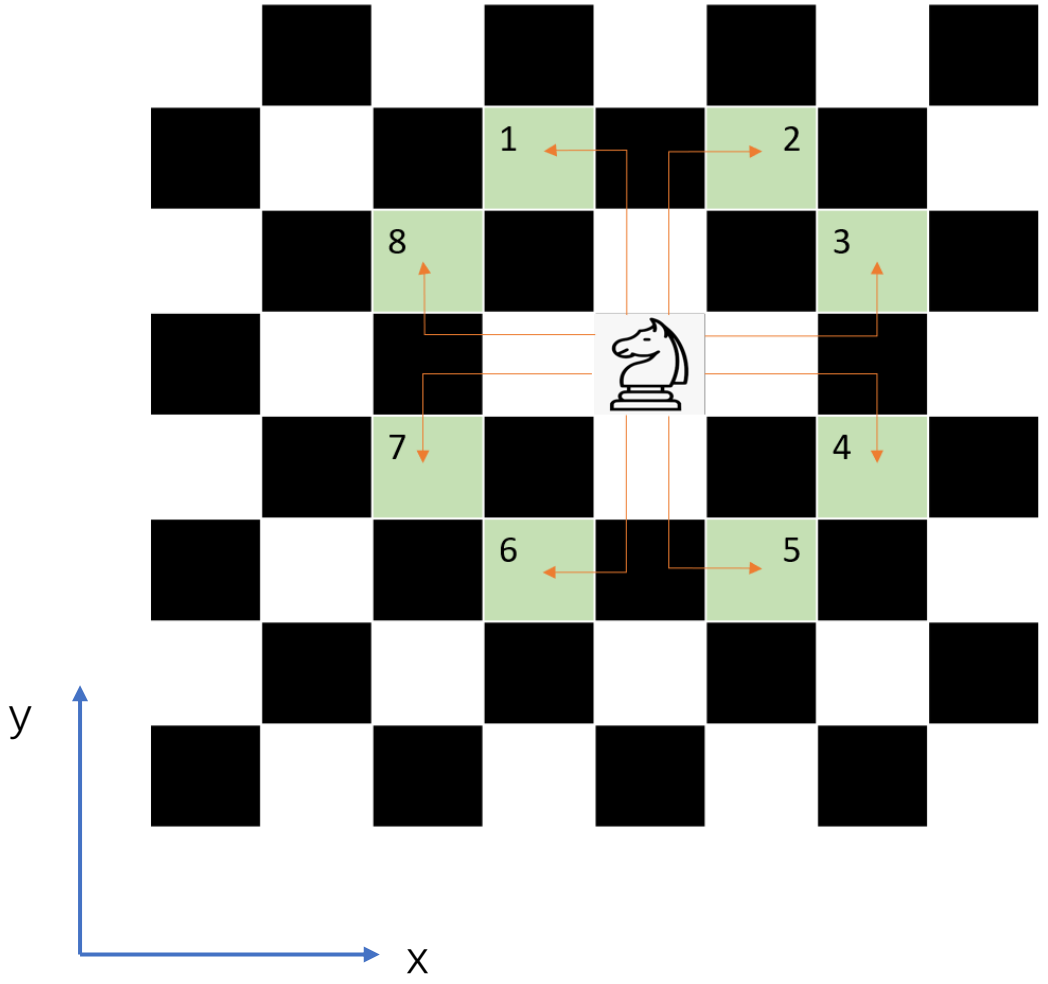


Reference: https://medium.com/nerd-for-tech/minimum-knight-moves-daily-challenge-may-4499e5217df5

Analysis

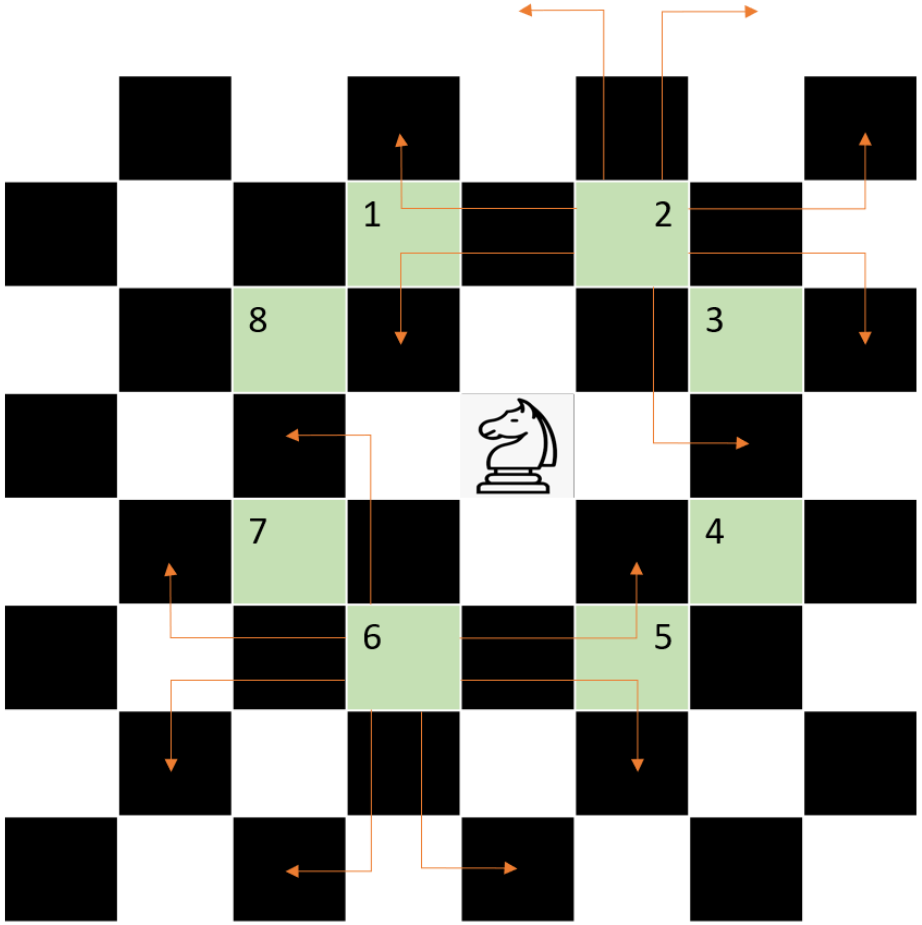
- Knight can move to 8 different positions

1. (x-1, y+2) ==> (-1, +2)
2. (x+1, y+2) ==> (+1, +2)
3. (x+2, y+1) ==> (+2, +1)
4. (x+2, y-1) ==> (+2, -1)
5. (x+1, y-2) ==> (+1, -2)
6. (x-1, y-2) ==> (-1, -2)
7. (x-2, y-1) ==> (-2, -1)
8. (x-2, y+1) ==> (-2, +1)



Analysis

- When it reaches position 2 (for example)
- From there it can move the next seven places.



Grading

- You can choose a problem to implement, the remaining one only describes the idea of solving the problem.
- Total point: 1