

รายงานการศึกษา

algorithm ของ n-queen problem

โดย

นาย ชญานิน เลียงจินดาถาวร (ไม้โมก) รหัสนักศึกษา 63010177

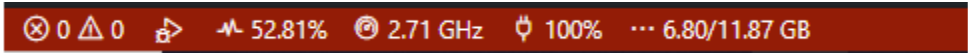
ทรัพยากร

Spec ของ เครื่องที่ใช้

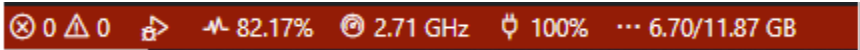
Device specifications

Device name	DESKTOP-GCM80HM
Processor	Intel(R) Core(TM) i5-7200U CPU @ 2.50GHz 2.70 GHz
Installed RAM	12.0 GB (11.9 GB usable)
Device ID	9E805D35-AD4B-42F8-A3BF-6CFA66145127
Product ID	00328-00000-00000-AA284
System type	64-bit operating system, x64-based processor
Pen and touch	No pen or touch input is available for this display

ขณะรันIterative



ขณะรันRecursion



ผลลัพธ์การทำงาน

Iterative

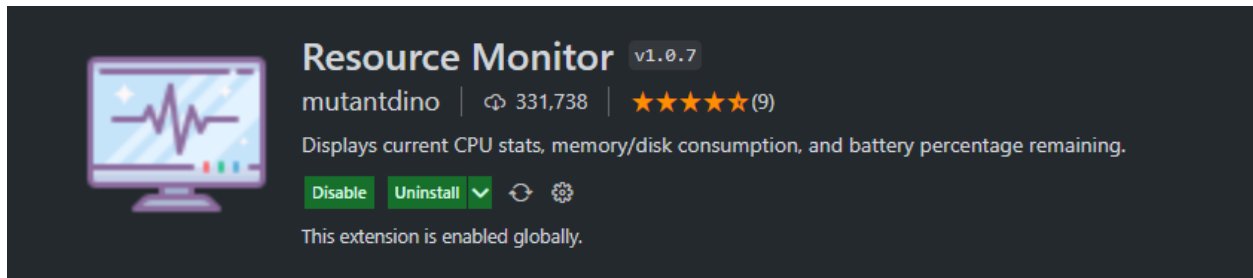
N = 4
time = 0.0006969000000000003
N = 5
time = 0.00481280000000000615
N = 6
time = 0.03162949999999998
N = 7
time = 0.3654995000000001
N = 8
time = 1.9863028
N = 9
time = 21.3438363
N = 10
time = 250.7711445

Recursion

N = 4
time = 0.000300600000000275977
N = 5
time = 0.00027249999999412466
N = 6
time = 0.001389299999971172
N = 7
time = 0.00411869999992259
N = 8
time = 0.012652899999977762
N = 9
time = 0.07938439999998081
N = 10
time = 0.23790780000001632

แหล่งอ้างอิง

ตรวจสอบทรัพยากรขณะรันโดย Extension Resource Monitor



โค้ดแบบ Iterative จาก

[Solve the 8 Queens Problem in Python | by Sergio Lopez |](#)
[Python in Plain English](#)

โค้ดแบบ Recursion จาก อาจารย์

Module time จาก

[time — Time access and conversions — Python 3.10.0](#)
[documentation](#)

วาดกราฟจาก Desmos

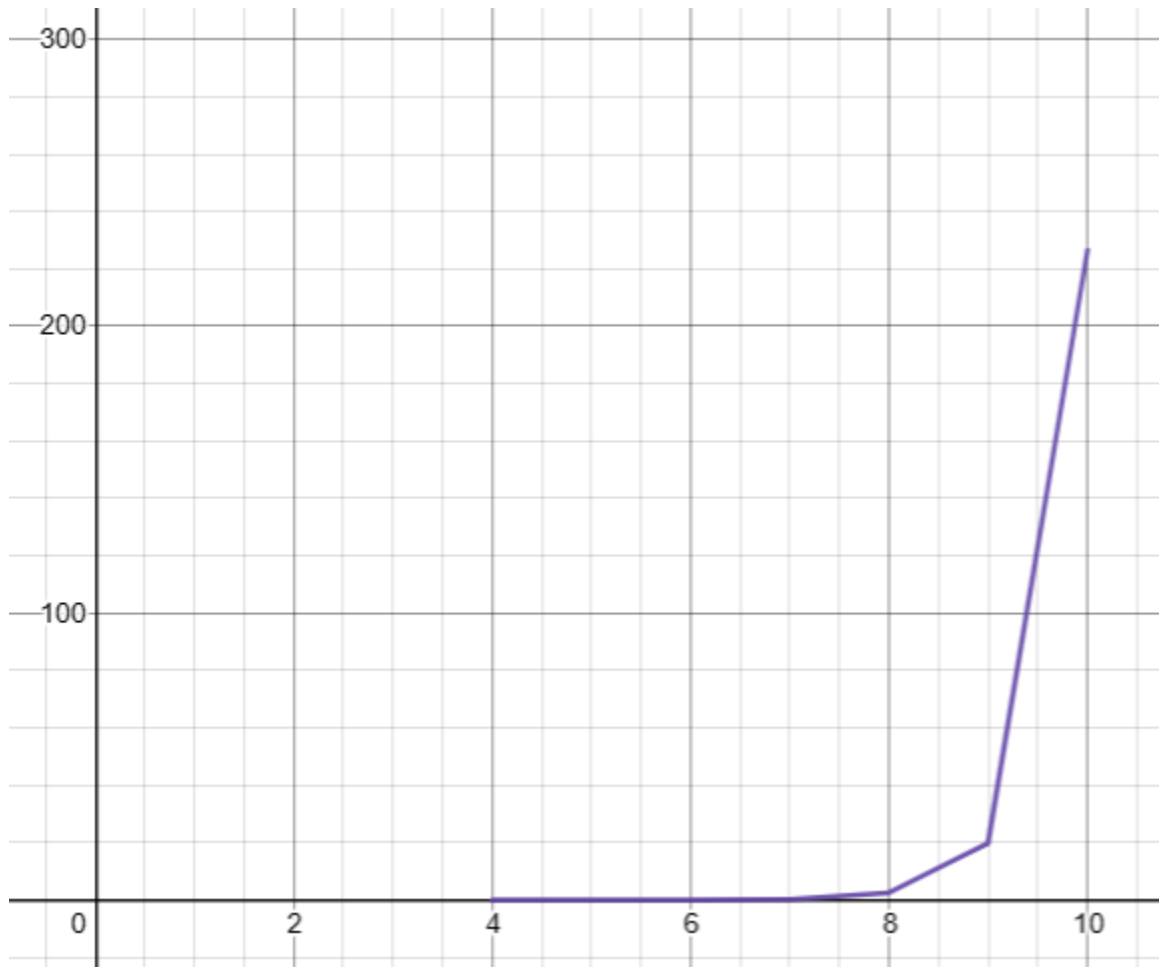
[เครื่องคิดเลขกราฟิก \(desmos.com\)](#)

ตารางบันทึกผล กราฟเปรียบเทียบ และสรุปการทดลอง

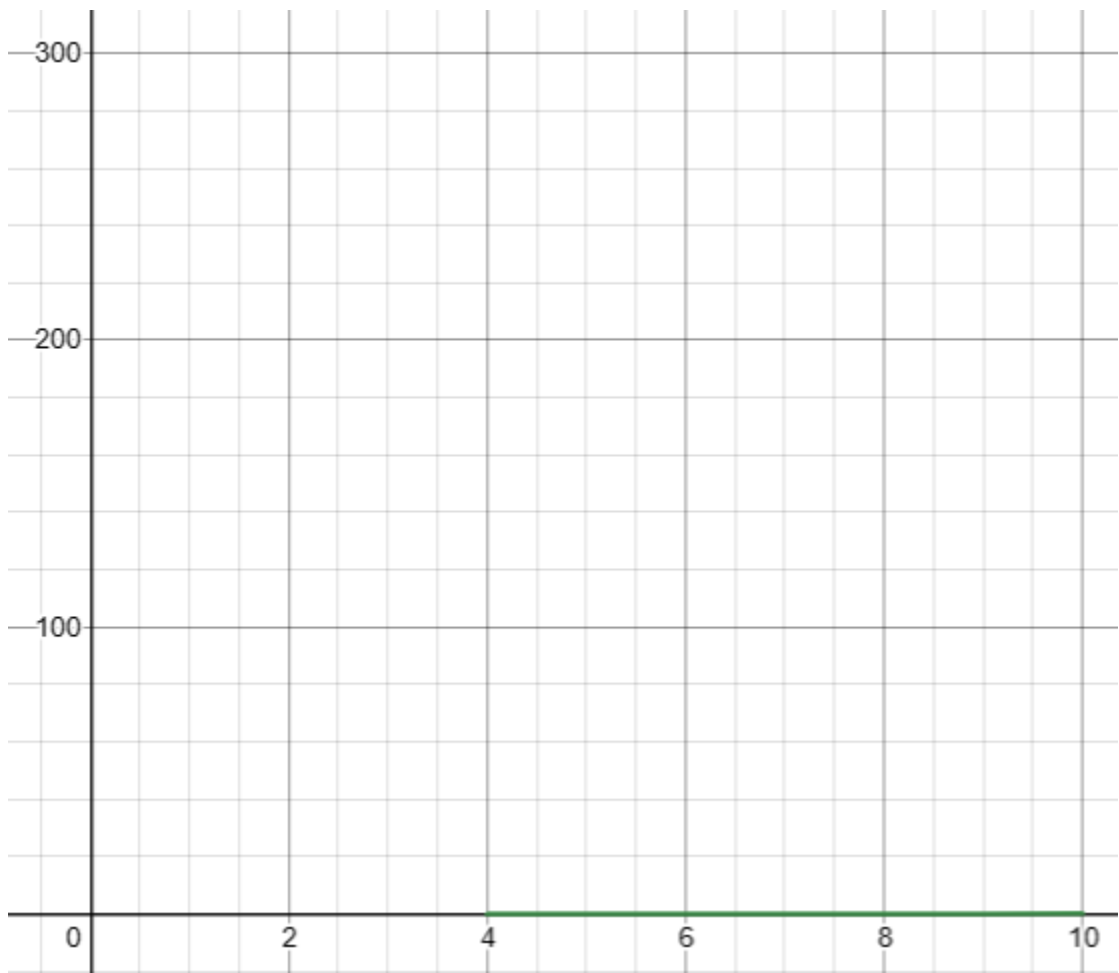
ตารางบันทึกผลเวลาการทำงานของ n-queen algorithm
(วินาที)

N	Iterative	Recursion
4	0.0010379	0.0002156
5	0.004623	0.0004849
6	0.0399532	0.0006922
7	0.2745237	0.0047337
8	2.5177999	0.0106132
9	19.9316482	0.0397327
10	226.5292748	0.1698758

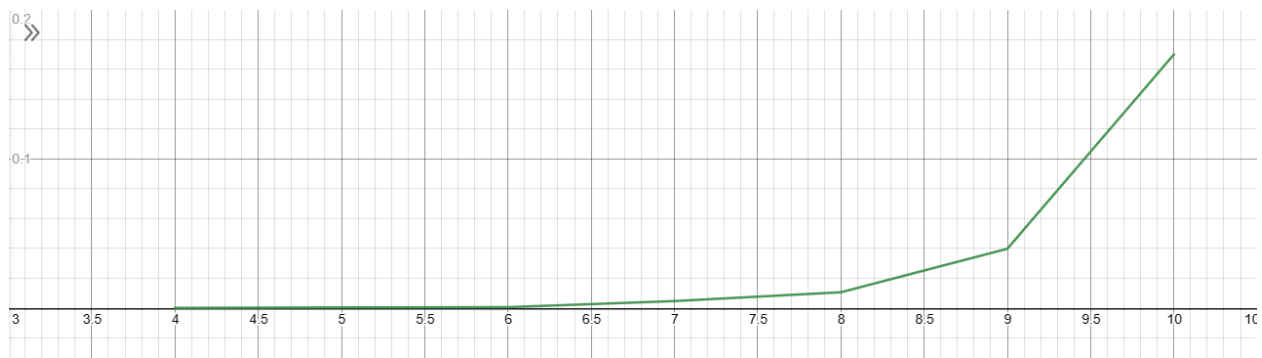
กราฟแสดงการทำงานแบบiterative(วินาที)



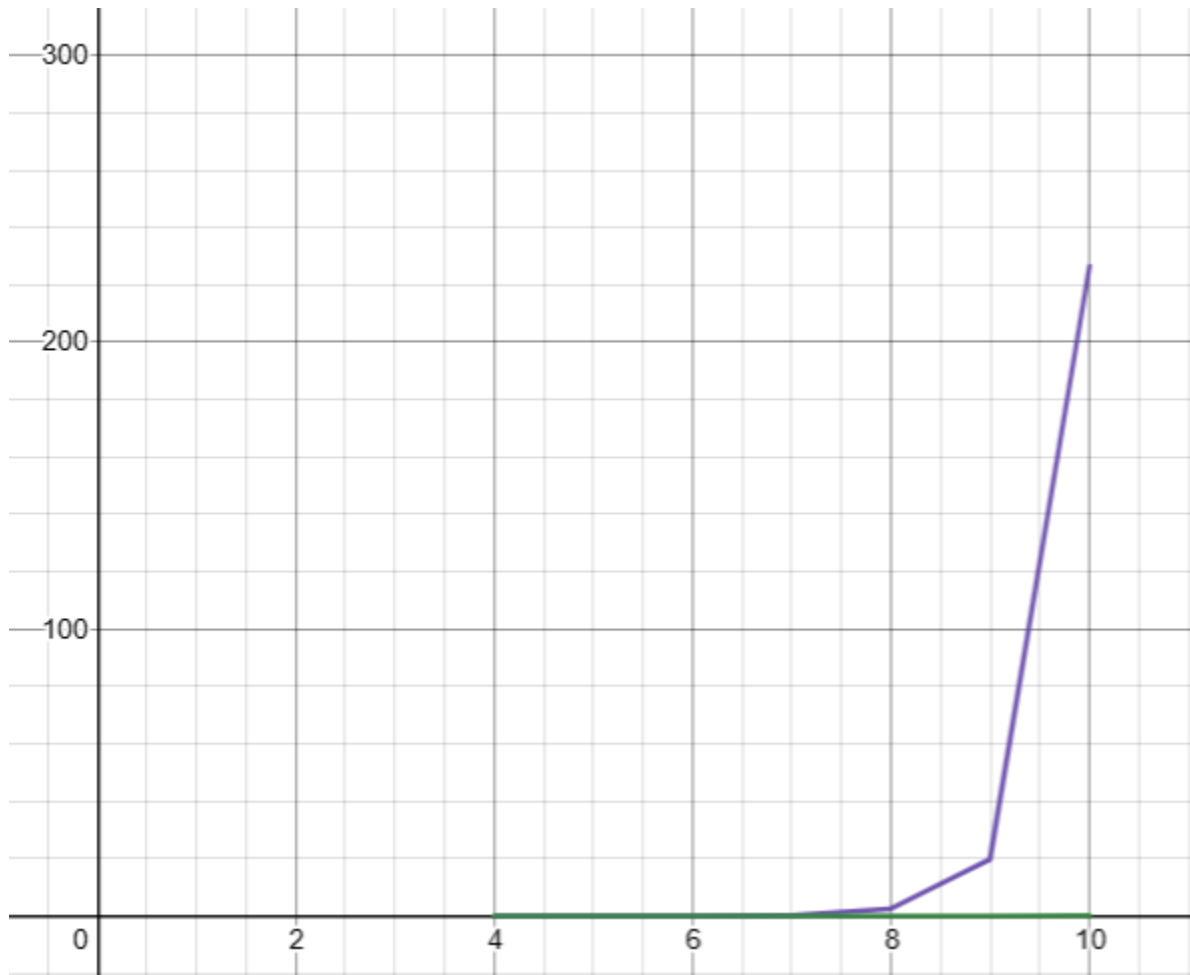
กราฟแสดงการทำงานแบบRecursion(วินาที)



ปรับสเกลเพื่อให้เห็นความแตกต่าง



กราฟเปรียบเทียบการทำงานทั้ง2แบบ(วินาที)



สรุปผล

จากการทดลองทำให้ได้เวลาการทำงานของalgorithm n-queen จากทั้ง 2 วิธีคือ Iterative และ Recursion ซึ่งผลลัพธ์ของทั้ง 2 วิธีทำให้ได้กราฟในลักษณะของ Exponential แต่ในการทำงานแบบ Iterativeมีการทำงานที่ช้ากว่ามากเนื่องจากการทำงานของ Iterative นั้นจะทำงานในทุกๆช่องที่เป็นไปได้และทุกรูปแบบโดยสามารถวิเคราะห์เป็น $O(n!)$

(<https://python.plainenglish.io/coding-the-8-queens-problem-in-python-d168f8df844b>)

ส่วนการทำงานของ Recursion นั้นอาศัยการ backtracking ทำให้ใช้เวลาในการทำงานน้อยกว่า แต่ทำให้เกิดการ Stack ขึ้นทำให้มีการใช้ CPU มากกว่าการทำงานแบบ Iterative