姓名：許哲瑜

For the hill climbing algorithm, generate 2 or 8 solutions for 4, 8, 16, 32 queens problems, write the total generated nodes and boards. Calculate the average searched nodes and stuck rate.

Stuck rate = (Total generated boards - #solutions) / Total generated boards

Where the solution is the optimal solution(No queen threats each other), the generated boards is the suboptimal solutions(Some queens threats each other).

**1. Find those # solutions, duplicated boards are ok.**

**a. hill climbing with random restart**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| size | #solutions | Total  Generated  nodes | Total  Generated  final boards | average searched nodes for one board | stuck rate | Execution time |
| 4 | 2 | **25** | **9** | **25/9=** **2.77778** | **(9-2)/9=**  **0.7778** | **13ms** |
| 8 | 10 | **303** | **86** | **3.52326** | **0.883721** | **213ms** |
| 16 | 10 | **3944** | **663** | **5.94872** | **0.984917** | **3102ms** |
| 32 | 10 | **17190** | **1484** | **11.5836** | **0.993261** | **51.826s** |

**b. local beam with k state, you could choose a suitable k yourself.**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| size | #solutions | Total  Generated  nodes | Total  Generated  final boards | average searched nodes for one board | stuck rate | Execution time |
| 4  k=2 | 2 | **16** | **2** | **8** | **0** | **2ms** |
| 8  k=4 | 10 | **1392** | **10** | **139.2** | **0** | **84ms** |
| 16  k=8 | 10 | **4843** | **11** | **440.273** | **0.090909** | **1577ms** |
| 32  k=16 | 10 | **2068** | **10** | **206.8** | **0** | **5996ms** |

**2. Find those # solutions without duplicated boards.The tables are the same as question 1.**

**a. hill climbing with random restart**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| size | #solutions | Total  Generated  nodes | Total  Generated  final boards | average searched nodes for one board | stuck rate | Execution time |
| 4 | 2 | **47** | **17** | **2.76471** | **0.882353** | **24ms** |
| 8 | 10 | **308** | **89** | **3.46067** | **0.88764** | **195ms** |
| 16 | 10 | **4595** | **954** | **6.09416** | **0.986737** | **3480ms** |
| 32 | 10 | **23354** | **2029** | **11.5101** | **0.995071** | **71119ms** |

**b. local beam with k state, you could choose a suitable k yourself.**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| size | #solutions | Total  Generated  nodes | Total  Generated  final boards | average searched nodes for one board | stuck rate | Execution time |
| 4  k=1 | 2 | **12** | **2** | **6** | **0** | **3ms** |
| 8  k=4 | 10 | **6244** | **11** | **567.636** | **0.0909** | **326ms** |
| 16  k=8 | 10 | **15856** | **10** | **1585.6** | **0** | **4983ms** |
| 32  k=2 | 10 | **28370** | **16** | **1773.12** | **0.3375** | **76494ms** |

討論：

1. 兩種方法的比較，ex. 搜尋速度，搜尋成功率…(Compare hill climbing and local beam search for search speed, sueecssful rate, and so on...)

Ans: hill climbing是隨機找一個點開始搜尋，相較於local beam search使用K值多點幫助搜尋，產生的nodes比較多，stuck rate較高成功率低，程式執行總時間也比較長。

local beam search相對來說各項數據都較好看。

2. K值的設定對搜尋的影響(How the K affect the local beam search?) K值的設定值越大代表留越多的node會繼續搜尋下去，更容易找queens problems的最佳解，也會使stuck rate大幅降低成功率變高，執行時間也越快。

Q&A

1.作業1是要求產出10個有效且不重複的解嗎?那作業表格上的 #solutions 是該尺寸的總有效解數目嗎?還是需要我們自行修改?

2.表格中的total generated nodes 是指產出10個有效不重複答案過程總耗費的board數目嗎?

3.表格中的total generated final boards 是指題目要求的10個有效答案嗎?

ANSWER:

1. #solutions 只要有跑出來 最佳解就算，不用管重複

2. total generated nodes是所有搜尋所產生node的個數

3. Total generated final boards 是每次搜尋截止的次數，包括最佳解和次佳解(近似解)

4. 最佳解是此solution皇后之間沒有威脅，次佳解(近似解)是此solution皇后之間有威脅