quickfind – weight quickfind Percolation report

b06505004 莊博翰

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Describe how you implemented Percolation.java. How did you check

\* whether the system percolates?

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*基本上如果要描述一組點是連接的可以將其想像成一個set，連接的點會放在同一個set中，所以實作上會產生一個quickunion ，每多一個點會依序將它所在的set 與上下左右open的點所在的set做union，所有最上方一列的點會與一個叫top的點union ，最下方一列的點會與一個叫ground 的點union，如此一來只要確認top與ground 是否在一個set 中就知道上下是否連通。

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

computational experiments n=k T=100 for Quickunion

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| n | 16 | 24 | 36 | 54 | 81 | 121 | 182 | 273 |
| time | 0.017 | 0.021 | 0.035 | 0.081 | 0.178 | 0.529 | 1.83 | 6 |
| ratio(t/n) | 0.001063 | 0.000875 | 0.000972 | 0.0015 | 0.002198 | 0.004372 | 0.010055 | 0.021978 |
| log(ratio) | -2.97367 | -3.05799 | -3.01223 | -2.82391 | -2.65807 | -2.35933 | -1.99762 | -1.65801 |

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

computational experiments n=100 T=k for Quickunion

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| T | 20 | 40 | 80 | 160 | 320 |
| times | 0.069 | 0.135 | 0.272 | 0.439 | 0.932 |
| ratio(t/n) | 0.00345 | 0.003375 | 0.0034 | 0.002744 | 0.002913 |
| log(ratio) | -2.46218 | -2.47173 | -2.46852 | -2.56166 | -2.53573 |

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

formula of n T for Quickunion

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

觀察n-times 的趨勢線為O(n^3 )，

time=3E-07x3 + 7E-06x2 - 0.0005x + 0.0296 (x=n , T=100)

觀察 T-times 的趨勢線為O(n) ，也就是T 與time 呈正比

將T加入上式:

time=(3E-07n3 + 7E-06n2 - 0.0005n + 0.0296)\*(T/100)

=

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

computational experiments n=k T=100 for weightQuickunion

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| n | 16 | 24 | 36 | 54 | 81 | 121 | 182 | 273 |
| time | 0.021 | 0.018 | 0.024 | 0.051 | 0.072 | 0.129 | 0.27 | 0.59 |
| ratio | 0.001313 | 0.00075 | 0.000667 | 0.000944 | 0.000889 | 0.001066 | 0.001484 | 0.002161 |
| log(ratio) | -2.8819 | -3.12494 | -3.17609 | -3.02482 | -3.05115 | -2.9722 | -2.82871 | -2.66531 |

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

computational experiments n=100 T=k for weightQuickunion

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| T | 20 | 40 | 80 | 160 | 320 |
| times | 0.045 | 0.066 | 0.144 | 0.194 | 0.308 |
| ratio(t/n) | 0.00225 | 0.00165 | 0.0018 | 0.001213 | 0.000963 |
| log(ratio) | -2.64782 | -2.78252 | -2.74473 | -2.91632 | -3.0166 |
|  |  |  |  |  |  |

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

formula of n T for weightQuickunion

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

觀察n-times 的趨勢線為O(n^3 )，

time= 6E-09x3 + 5E-06x2 + 0.0003x + 0.0112 (x=n , T=100)

觀察 T-times 的趨勢線為O(n) ，也就是T 與time 呈正比

將T加入上式:

time=(6E-09x3 + 5E-06x2 + 0.0003x + 0.0112)\*(T/100)

=

相較quickunion 三次項少了許多 實際時間複雜度因該介於O(n^3)到 O(n^2)之間，猜測為O(n^2\*log(n))

handle backwash

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

我用了簡單暴力的方法，多另一個quickunion2 一樣上方接top但底下不接ground用原本的quickunion1(接top 接ground)判斷是否連通，用quickunion2

來繪製圖形，其餘操作quickunion1 ,quickunion2同步執行。如此一來在繪製圖形時最下列不會因都連接著ground被視為連通。