Минимална описана окръжност

Проект по системи за паралелна обработка

Изготвили: Калин Маринов, 81051

Кристиана Неделчева, 81052

Проверил: ……………………………………………………….

/ас. Христо Христов/

# Условия на задачата

# Разглеждаме множество от точки в равнината. Точките имат целочислени координати в интервала [0, 47483647]. Съществуват безброй много окръжности в равнината, съдържащи това множество от точки.

# Вашата задача е да напишете програма намираща минималната, описана около множеството от точки окръжност. Програмата намира минималната описана окръжност използвайки паралелни процеси (нишки).

## аНАЛИЗ И СВОЙСТВА

* Съществува единствена минимална описана окръжност за дадено множество от точки
* Минималната окръжност винаги минава през две или три точки
  + В противен случай би съществувала друга покриваща окръжност с по-малък радиус
* При минимална окръжност инцидентна на две точки, тя има диаметър равен на разстоянието между точките
* Когато окръжността е дефинирана от три точки – тя представлява окръжността описана около триъгълника формиран от трите точки
* За всяка намерена окръжност през две точки, съдържаща всички останали – твърдим че тя е минимална.
  + Никоя окръжност с радиус по-малък от намерената не може да съдържа двете точки
  + Следователно при откриване такава окръжност можем да сме сигурни че тя е единственото решение.

## Инкрементално решение

Ще използваме итеративен подход, като на K-тата итерация ще намираме окръжността покриваща първите K на брой точки. Тази окръжност ще означим с Ck. Да разгледаме действието на алгоритъма на K-тата итерация:

Нека с означим първите точки и е намерената окръжност от предишната итерация, съдържаща това множество от точки.   
В зависимост от позицията на новата точка :

* Ако е вътрешна за , то (няма промяна в решението)
* В противен случай търсим нова окръжност за минаваща през точката

## Намиране на окръжност - нАИВЕН ПОДХОД

След като знаем че търсената окръжност минава през дадена точка , можем да създадем алгоритъм, който генерира всички окръжности през две и три точки, като една от тях е .

Така постигаме високо ниво на паралелност на програмата, въпреки че избрания подход намиране всички комбинации от окръжности.

На K-тата итерация, сложността на алгоритъма за обхождане на окръжности през и една от другите точки, включително проверка на условието за минималност на всяка окръжност е:

Сложността за генериране на всички окръжности и проверка за минималност през и две други точки е съответно:

Важно е да се отбележи че при намиране на една окръжност през две точки, съдържаща всички съществуващи точки – тя е минимална, следователно можем да прекратим търсенето. Същото обаче не може да се твърди за окръжност през три точки.

При K-тата итерация вероятността точката да е извън намерената окръжност от предходната итерация е: . В случай че точката е вътрешна за окръжността, не е необходимо търсенето на нова окръжност – решението се запазва и сложността е O(1).

Така общата сложността на алгоритъм с итерации е:

# Архитектура на приложението и реализация на алгоритъма

* Библиотеката SmallestCircle.Data се използва за
  + Итерация на точките (от файл или с генератор на произволни такива)
  + Моделиране на обектите точка и окръжност
* Библиотеката SmallestCircle.Calculation има за цел
  + Дефиниране на геометрични свойства – разстояние между точки, принадлежност към окръжност на точка или списък от точки
  + Изчисляване на окръжност – през две или три точки
  + Калкулация на минимална окръжност
    - Calculator class – калкулира минималната окръжност чрез итеративен подход, използва една нишка, без възможност за асинхронно четене на файла.
    - MultiCalculator class – Multi-threading и асинхронен вариант на алгоритъма от Calculator класа
    - DemoCalculator class – служи за единствено за демонстрация, принтиране на състоянието на нишките и онагледяване на алгоритъма на графичния интерфейс
  + SmallestCircle.Console – конзолна апликация за стартиране на проекта
  + SmallestCircle.Presentation – WPF апликация за демонстрация на работата на алгоритъма
    - Възможност за подваане на точките през визуален интерфейс
    - Възможност за експортиране на точките
    - Демонстрация на многонишковия калкукатор с възможност спиране и продължаване

# Проведени тестове и измервания

Докато чакахме да ни бъде предоставена тестова машина Windows проведохме няколко тестове на машините, с които разполагаме. На първата машина, на която проведохме тестове е с до 4 ядра.

|  |  |  |  |
| --- | --- | --- | --- |
| **Брой ядра на процесора** | **Време в ms** | **Ускорение** | **Ефективност** |
| 1 | 6052 ms |  |  |
| 2 | 3248ms |  |  |
| 3 | 2462 ms |  |  |
| 4 | 1908ms |  |  |

Just click to insert one of these and you’ll be prompted to update the TOC. When you do, text you formatted using Heading 1, Heading 2, and Heading 3 styles is automatically added.

|  |  |  |  |
| --- | --- | --- | --- |
| **Брой ядра на процесора** | **Време в ms** | **Ускорение** | **Ефективност** |
| 1 | 8801 ms |  |  |
| 2 | 5013 ms |  |  |
| 4 | 2615 ms |  |  |
| 8 | 1704 ms |  |  |
| 12 | 1262 ms |  |  |
| 14 | 1283 ms |  |  |
| 16 | 1190 ms |  |  |
| 24 | 1442 ms |  |  |

## Add a Bibliography

On the References tab, in the Citations & Bibliography group, click Insert Citation for the option to add sources and then place citations in the document.

When you’ve added all the citations you need for your report, on the References tab, click Bibliography to insert a formatted bibliography in your choice of styles.

And you’re done. Nice work!