

Rio-projekti: Kuvankäsittely

Tero Huomo, Eeva Terkki, Jussi Kirjavainen ja Niko Ahonen

25. maaliskuuta 2012

Sisältö

1 Johdanto	3
1.1 Alkusanat	3
1.2 Tehtävän määrittely	3
2 Ohjelma	4
2.1 Toimintaperiaate	4
2.2 Luokat	4
2.2.1 Bitop.java	4
2.2.2 ImageProcessingMain.java	4
2.2.3 PixelImage.java	4
2.2.4 PpmImageParser.java	4
3 Testaus	5
3.1 Testauksen päämäärät	5
3.2 Testiluokat	5
3.2.1 BitopTest.java	5
4 Yhteenveto	6
A Projects	7

Kuvat

Luku 1

Johdanto

1.1 Alkusanat

Teemu Kerola luennoi keväällä 2012 Helsingin yliopiston tietojenkäsittelytieteen laitoksella kurssin *Rinnakkaisohjelmointi*. Kolmasosa kurssin arvosanasta muodostuu pienryhmissä tehtävästä projektityöstä. Tämä työ on ryhmämme projektityön dokumentointi.

1.2 Tehtävän määrittely

Kuudesta mahdollisesta projektityön aiheesta (liite A) ryhmämme valitsi kuvankäsittelyyn liittyvän aiheen *Image processing for ppm images*. Tässä tehtävänä oli ensin sumentaa annettua ppm-muotoista kuvaa kolme kertaa ja sitten tarkentaa sitä kolme kertaa. Tehtävän tulosteesta piti ilmetä vähintään:

- Kuvankäsittelyyn kulunut aika
- Käytettyjen prosessorien ja säikeiden lukumäärä
- Prosessoitu kuva tallennettuna samalla formaatilla kuin alkuperäinen

Luku 2

Ohjelma

2.1 Toimintaperiaate

2.2 Luokat

2.2.1 Bitop.java

2.2.2 ImageProcessingMain.java

2.2.3 PixelImage.java

2.2.4 PpmImageParser.java

Luku 3

Testaus

3.1 Testauksen päämäärät

3.2 Testiluokat

3.2.1 BitopTest.java

Luku 4

Yhteenveto

Liite A

Projects

- **Sorting integers(to ascending order)**

- Minimum outputs:
 - * Time used to sort the integers
 - * Amount of threads/processors used
- Performance competition data set is */fs-1/2/kerola/rio_testdata/uint64-keys.bin* (64 bit little-endian unsigned integers, 58M values, file size 0.47 GB; read from file server, do not make extra copies and waste space...)
 - * In 2011 best time was 1,72 s (Java)
- Java does not support well 64 bit unsigned integers. You may interpret the data set as signed integers instead.

- **Sorting strings(to ascending order)**

- Minimum outputs:
 - * Time used to sort the strings
 - * Amount of threads/processors used
- Performance competition data set is */fs-1/2/kerola/rio_testdata/utf-8-keys.txt* (8 bit utf, 58M lines, file size 0.74 GB; read from file server, do not make extra copies and waste space...)
 - * In 2011 best time was 98 s (Java)

- **Matrix multiplication**

- Minimum outputs:
 - * Time used to multiply matrices
 - * Amount of threads/processors used
- Performance competition data set consists of text file matrixes
 - * */fs-1/2/kerola/rio_testdata/matr1_7500.txt*
 - * */fs-1/2/kerola/rio_testdata/matr2_7500.txt*
 - * In 2011 best time was 66,6 s (Java)
- Data set is given as two text files, each holding one matrix. Both files start with decimal integer N denoting the array size and followed by NxN array in row wise order (single accuracy floating points), values separated with white space. E.g.,
 - * file 1: 3 1.23 3.45 6.78 2.23 1.45 5.78 5.23 5.45 5.78
 - * file 2: 3 1.33 3.35 6.38 2.33 1.35 5.38 5.33 5.35 5.38
- Tommi Tuura made 2 practice matrices for testing purposes. These files are only to aid the development of your program, they are not substitutes for the larger matrices in any way:
 - * */fs-1/2/kerola/rio_testdata/matr1_1000.txt*
 - * */fs-1/2/kerola/rio_testdata/matr2_1000.txt*

- **Game of life**

- Minimum outputs:
 - * Time used to evolve all steps of life from the grid
 - * Amount of threads/processors used
 - * Final state of the grid saved in a file in same format as competition data set (in this file, the second integer should denote the number of steps that have been run)
- Performance competition data set (grid size NxN, nr of steps P, starting state) is */fs-1/2/kerola/rio_testdata/life_800_10000.txt* (800x800 matrix, 10000 steps).
 - * Best time was 17,5 s (Java)

- Data set is given as a text file with integers N, P, NxN grid in row wise order (1 = alive, 0 = dead), values separated with white space

* E.g., 4 20 0 0 0 0 0 1 0 0 0 0 1 0 0 0 0 0

- **Image processing for ppm images**

- Minimum outputs:
 - * Time used to process the image as required
 - * Amount of threads/processors used
 - * The processed final result image saved in a file in same format as competition data set
- Performance competition data set is */fs - 1/2/kerola/rio_testdata/7976x4480.ppm* with operations smoothen, smoothen, smoothen, sharpen, sharpen, sharpen.
 - * in 2011 best time was 2,1 s (Java)

- **n-Queens (find all solutions)**

- Minimum outputs:
 - * Time used to calculate the positions for given n
 - * Amount of threads/processors used
 - * Calculated total amount of positions for given n (NO individual positions)
- Performance competition data set (problem size n) is 17.
 - * in 2011 best time was 144 s (Java)

Kirjallisuutta

- [1] TÄHÄN SITTEEN MEIDÄN TYÖSSÄ KÄYTETTÄVÄT LÄHTEET!
SAMAAN TYYLIIN KU NOI, JOTKA OTETAAN SITTEEN POIS...
- [2] Casey, John (1886). A sequel to the first six books of the Elements of Euclid.
Luettu 22.9.2011. Saatavissa: <http://www.archive.org/details/americana>
- [3] Coolidge, Julian (1916). A Treatise on the Circle and the Sphere. Luettu 21.9.2011.
Saatavissa: <http://www.archive.org/stream/treatiseoncircle00cooluoft#page/166/mode/2up/search/apollonius>
- [4] Dörrie, Heinrich (1965). 100 Great Problems of Elementary Mathematics: Their History and Solutions, sivut: 154-160. Luettu 6.10.2011.
Saatavissa: <http://www.scribd.com/doc/25259960/100-Great-Problems-of-Elementary-Mathematics-Dover-Tqw-darksider>