2023_Teme_lucrare

Sa se scrie o aplicatie Java care sa realizeze o procesare de imagini.

Cerintele temei:

- Tema este un proiect
- Lucrarea trebuie sa contina toate elementele unui proiect: introducere, descrierea aplicatiei cerute, partea teoretica, descrierea implementarii, descrierea structurala – architecturala si functionala a aplicatiei implementate, descrierea modulelor, evaluare performante, concluzii, bibliografie, documentatie cod sursa.
- Incercati o distributie a temelor in fiecare semigrupa. Nu sunt acceptabile lucrari care seamana prin continut (chiar partial).
- Nu este necesara interfata grafica

Teme de Procesare:

1. Converting Color Image to Gray-Scale Image – Average method
2. Normalize colors
3. Negative Image
4. Level Adjustment (contrast and black/white adjustment)
5. Sobel Operator
6. Prewitt Operator
7. Image Sharpening (convolution mask)
8. Image Smoothing (convolution mask)
9. Convert Gray-Scale Image to Binary image (Static Threshold)
10. Image mirroring
11. Binary Operation (AND, OR, XOR) between two images
12. Gray Level Histogram of a Gray-Scale Image (Imaginea rezultata este imaginea continand 11 niveluri de gri – toti pixelii care au nivelul de gri +/- 5 fata de nivelul de gri al maximului histogramei).
13. Decrease color depth Gray-Scale Image
14. Rotate Image (90, 180, 270)
15. Translate Image (X – Horizontal, Y – Vertical – prescribed by user)
16. Edge Extraction (Detection)
17. Converting Color Image to Gray-Scale Image – Weighted method (luminosity method)
18. Image resizing (Zooming +/-) – keeping aspect ratio. Pixel replication method
19. Image resizing (Zooming +/-) – keeping aspect ratio. Zero order hold method
20. Image resizing (Zooming +/-) – keeping aspect ratio. Zooming K times method
21. Image Brightness modification
22. Image Contrast modification
23. Gray Level Histogram Sliding (+/-)
24. Gray Level Histogram Stretching (+/-)
25. Linear Gray Level Transform
26. Logarithmic Gray Level Transform
27. Power-Law Gray Level Transform
28. Laplacian Operator (Positive/Negative)

Cerintele de implementare:

1. Imaginea sursa este BMP (fisier) – 24bit BMP – RGB
2. Pentru procesare se folosesc doar algoritmi si/ sau secvente de cod low-level (nu se accepta utilizare de metode de procesare altele decat cele scrise in tema)
3. Include in totalitate conceptele POO – incapsulare, mostenire, polimorphism, abstractizare
4. Codul sursa respecta absolut toate “Coding standards”. Codul sursa este comentat
5. Operatii de lucru cu fisiere
6. Operatii de intrare de la tastatura si prin parametri liniei de comanda pentru asignarea fisierelor de intrare, parametri / setarile / optiunile de executie si pentru asignarea fisierelor de iesire
7. Aplicatia trebuie sa fie multimodulara (impartirea in clase cu ierarhii – chiar cu cost in timp de procesare). Cel putin 3 niveluri de mostenire
8. Include varargs
9. Include constructori
10. Include cel putin un bloc de initializare si un bloc static de initializare
11. Include Interface (cu o clasa care o implementeaza)
12. Include Clase Abstracte cu metode abstracte si clase concrete care extind clasele abstracte
13. Include tratarea exceptiilor
14. Aplicatia contine 2 pachete: Pachetul 1 sa contina aplicatia de test, pachetul 2 sa contina restul claselor

15. Aplicatia contine Producer-Consumer cu urmatoarele cerinte:
    a. un nou thread este alocat citirii din fisier a imaginii sursa – Producer Thread. Intra in Not Runnable dupa citirea a unui segment de informatie
    b. un nou thread (Consumer Thread) este alocat consumului informatiei furnizate de Producer Thread. Se utilizeaza “multithread communication” (notify).
    c. Se insereaza output la consola si sleep (1000) pentru a evidentia etapele comunicarii.
    d. Se folosesc elementele de sincronizare pentru protectia la o eventuala interferenta cu alte posibile threaduri
    e. Dupa terminarea consumului intregii informatii de imagine sursa, se incepe procesarea

16. Aplicatia contine comunicatie prin Pipes cu urmatoarele cerinte
    a. Consumer utilizeaza un Pipe pentru a transmite imaginea procesata catre un obiect de tipul WriterResult
    b. Transmiterea prin pipe se face partitionand informatia in 4 segmente.
    c. La transmiterea fiecarui segment segment se trimite la consola un mesaj
    d. La receptia fiecarui segment segment se trimite la consola un mesaj
    e. Rezutatul se depune intr-un fisier

Cerintele 15 si 16 au ponderea mai mare fata de celelalte cerinte.

Etapele de executie ale aplicatiei sunt:

- citire informatii de identificare fisier sursa (fisiere sursa) si citire informatii de identificare fisier destinatie
- citire fisier sursa
- procesare imagine
- scriere fisier destinatie
- inregistrare timp de executie fiecare etapa
- afisare rezultate timp de procesare fiecare etapa

Nota: Nota obtinuta la proiect va reflecta respectarea tuturor cerintelor temei, cerintelor de implementare si a etapelor de executie