

## TEMA 3 - 4 saptamani

### Aplicatie JavaScript care sa realizeze o procesare de imagini.

#### Cerintele temei:

- Tema este un proiect
- Documentul lucrarii trebuie sa contina toate elementele unui proiect: descrierea aplicatiei cerute, partea teoretica, descrierea implementarii, descrierea functionala a aplicatiei implementate, descrierea modulelor, bibliografie, comentare cod sursa.

#### Cerintele de implementare:

1. Imaginea sursa este continuta intr-un JSON obtinut de la API DOG (<https://dog.ceo/dog-api/>)
2. Componentele JSON-ului obtinut se afiseaza in browser
3. Imaginea din JSON se prelucreaza in canvas. Pentru prelucrarea imaginii exista urmatoarele optiuni:

**Optiunea 1:** un mirror (*astfel incat pixelii din dreapta sa ajunga in stanga si invers*) urmata de prelucrarea imaginii rezultate conform **temei de procesare** distribuita

sau

**Optiunea 2:** aplicarea prelucrarii pe jumatatea din *stanga* a imaginii urmata de prelucrarea imaginii rezultate conform **temei de procesare** distribuita

4. Scrierea in canvas se face dupa un timp prestabilit prin utilizarea **setTimeout**.
5. Pentru procesare se folosesc doar algoritmi si/ sau secvente de cod low-level (nu se accepta utilizarea de metode de procesare altele decat cele scrise in tema)
6. Codul sursa respecta "Coding standards". Codul sursa este comentat
7. Prelucarea JSON se face asincron (**async, await**)
8. Procesarea imaginii din canvas se face asincron impartind actiunea de procesare in 4 felii de executie (cu un timp intre ele de 1s)

#### Etapele de executie ale aplicatiei sunt:

- Afisare componente JSON in browser
- procesare imagine
- inregistrare timp de executie fiecare etapa
- afisare rezultate timp de procesare fiecare etapa

## **Teme de Procesare:**

1. Converting Color Image to Gray-Scale Image – Average method
2. Normalize colors
3. Negative Image (inversare de imagine pe  $\frac{1}{2}$  atat pe verticala cat si pe orizontala)
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 (atat pe verticala cat si pe orizontala)
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)

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