

Laborator 5

1 Obiective

Obiectivul acestui laborator este de a folosi operațiile cu vectori și matrici definite în laboratoarele anterioare pentru a efectua diferite transformări pe o figură 2D.

2 Utilizarea aplicației SDL

Pentru exemplificarea grafică a transformărilor 2D, vom folosi o aplicație foarte asemănătoare cu cea din laboratorul 1, bazată pe biblioteca SDL (Simple DirectMedia Layer).

3 SDL Renderer

SDL_Renderer este o structură care gestionează procesul de rasterizare, legată de un **SDL_Window**. Un obiect de tip **SDL_Renderer** poate să țină evidența setărilor de rasterizare.

Există câteva funcții importante asociate obiectelor de tip **SDL_Renderer**:

1. **SDL_SetRenderDrawColor** stabilește culoarea care va fi utilizată în toate operațiile de rasterizare, până când se efectuează un alt apel la funcție.

```
SDL_SetRenderDrawColor(renderer, r, g, b, a);
```

2. **SDL_RenderClear** șterge întreaga zonă de afișare a ferestrei utilizând culoarea activă curentă, setată anterior folosind funcția **SDL_SetRenderDrawColor**.

```
SDL_RenderClear(renderer);
```

3. **SDL_RenderPresent** va afișa întregul conținut rasterizat pe suprafața ecranului. Până când această funcție este apelată, procesul de rasterizare are loc într-un buffer ascuns care nu este vizibil pentru utilizator. Apelul **SDL_RenderPresent** trebuie efectuat o singură dată, după ce toate funcțiile de rasterizare au fost apelate.

```
SDL_RenderPresent(renderer);
```

Dacă dorim să desenăm un dreptunghi, trebuie să specificăm poziția de pornire a dreptunghiului (coordonatele x și y ale colțului stânga-sus) și dimensiunea (lățimea și înălțimea). Pentru a specifica o culoare folosim structura **SDL_Color** utilizând valorile de culoare roșu, verde, albastru și alfa. Funcția **SDL_RenderFillRect()** va desena dreptunghiului folosind culoarea setată anterior.

```
SDL_FRect rectangleCoordinates = {100, 100, 200, 200};
SDL_Color rectagleColor { 255, 0, 0, 255 };
SDL_SetRenderDrawColor(renderer,
    rectagleColor.r, rectagleColor.g, rectagleColor.b, rectagleColor.a);
```

```
SDL_RenderFillRect(renderer, &rectangleCoordinates);
```

Pentru fiecare canal de culoare specificăm valorile cuprinse între 0 și 255. În tabelul următor se regăsesc câteva exemple de culori care se pot obține folosind diferite combinații de valori ale celor trei canale.

Nume	Canal Roșu	Canal Verde	Canal Albastru	Culoare
Alb	255	255	255	
Roșu	255	0	0	
Verde	0	255	0	
Albastru	0	0	255	
Galben	255	255	0	
Negru	0	0	0	

Dacă dorim să desenăm o linie, trebuie să specificăm punctul de pornire și cel de sfârșit al liniei.

```
vec3 p1(100, 100, 1), p2(400, 100, 1);  
  
SDL_RenderLine(renderer, p1.x, p1.y, p2.x, p2.y);
```

4 Temă

Descărcați și rulați aplicația de pe site-ul web al laboratorului. Încercați să înțelegeți exemplul de bază și apoi extindeți aplicația cu următoarele funcționalități:

- Includeți în aplicație fișierele de implementare (.cpp).
- Definiți un dreptunghi inițial având colțul din stânga-sus în P1(100, 100) și colțul din dreapta jos în P2(400, 200).
- Rotiți dreptunghiul în jurul centrului său (intersecția diagonalelor) cu 10 grade în sensul acelor de ceasornic atunci când este apăsat RIGHT_ARROW și 10 grade în sens invers acelor de ceasornic atunci când este apăsat LEFT_ARROW.
- Aplicați o scalare dreptunghiul având colțul din stânga sus ca referință, utilizând tastele UP_ARROW și DOWN_ARROW.