

Programowanie interaktywnej grafiki  
komputerowej

**Lab 8: Transformacje, animacje**

# 1 Cel ćwiczeń laboratoryjnych

Celem ćwiczeń laboratoryjnych jest zapoznanie studenta z podstawowymi operacjami modyfikującymi układ współrzędnych Canvas oraz wprowadzeniem do animacji.

## 2 Podstawy obsługi płótna - Canvas

Przetestuj działanie poniższych funkcji umożliwiających modyfikację układu współrzędnych

- `translate(tx, ty)`,
- `rotate(alpha)`,
- `scale(sx, xy)`,

## 3 Zadanie wykres 1

Wykonaj wykres funkcji na Canvas wykorzystując do skalowania funkcje transformacji

$$y = \sin(x) + 0.5 * x - c \quad (1)$$

Jakie wady a jakie zalety ma takie rozwiązanie.

## 4 Zadanie wykres 2

Wykonaj wykres funkcji na Canvas realizując skalowanie z dziedziny danych funkcji do dziedziny danych Canvas bez użycia funkcji transformacji (funkcje transformacji mogą być użyte do odbicia układu współrzędnych - względem osi OX)

$$y = \sin(2x) * \cos(3x) + 0.5 * x - c \quad (2)$$

Jakie wady a jakie zalety ma takie rozwiązanie.

## 5 Animacja piłki

Wykonaj animację piłki odbijającej się od krawędzi Canvas zgodnie z szablonem animacji:

```
1     <script>
2     window.onload = function () {
3     var canvas = document.getElementById("AnimationCanvas");
```

```
4     var context = canvas.getContext("2d");
5
6     //Deklaracja zmiennych podlegających zmianie podczas całej animacji
7     var last_time, stop, linear_speed;
8
9     //Inicjalizacja stanu początkowego animacji
10    function InitAnimation()
11    {
12        stop = false;
13        var date = new Date();
14        last_time = date.getTime();
15        //Inicjalizacja zmiennych opisujących animację
16
17        //Uruchomienie animacji
18        window.requestAnimationFrame(drawAnimation);
19    }
20
21    //właściwa funkcja rysująca kolejne klatki animacji
22    function drawAnimation()
23    {
24        // 1. czyszczenie płótna
25        context.clearRect(0, 0, 500, 100);
26
27        // 3. wyznaczenie upływu czasu od ostatniej klatki
28        var date = new Date();
29        var time_interval = date.getTime() - last_time;
30
31        // 3.1. Wyznaczenie nowego stanu rysowanych obiektów
32        var distance = linear_speed * time_interval / 1000;
33
34        // 4. Rysowanie obiektów
35
36        // 4.1. Zapamiętanie stanu płótna
37        context.save();
38        context.beginPath();
39
40        // 4.2 Rysowanie obiektów
41        context.closePath();
42        context.restore();
43
44        // 5. ponowne wywołanie pętli animacji
45        if (!stop) window.requestAnimationFrame(draw);
46    }
47
48    InitAnimation();
```

```
49     }  
50   </script>
```

Do animacji piłki wykorzystaj funkcje transformacji.

## 6 Zadanie wykres kołowy

Zmodyfikuj zadanie wykresu kołowego z poprzednich zajęć laboratoryjnych tak, aby wykres był wyświetlany po wczytaniu w formie animacji polegającej na stopniowym rozwijaniu koła (poszerzaniu wycinków) od wartości zerowej, do wartości docelowej reprezentującej przykładowe dane. Dobierz właściwie czas początkowej animacji (zadanie podlega ocenie).

## 7 Zadanie wykres słupkowy

Narysuj wykres słupkowy, reprezentujący przykładowe dane opisane w formacie json

```
1  var dane_wykresu = {  
2      serie: [[10, 30, 45, 24, 79], [24, 15, 33, 34, 50]],  
3      etykiety: ["label1", "label2", "label3", "label4", "label5"],  
4      tytul: "Animowany_wykres_słupkowy",  
5      type: "wykres_słupkowy"  
6  };
```

Do rysowania należy wykorzystać wyłącznie Canvas.

Podczas pokazywania wykresu słupkowego wykonaj animację polegającą na stopniowym zwiększaniu wysokości słupków od wartości zerowej do wartości docelowej zawartej w źródle danych dla wykresu (zadanie podlega ocenie).