**SPRAWOZDANIE**

Zajęcia: Grafika komputerowa

Prowadzący: mgr Mikołaj Grygiel

**Laboratorium 5**

14.04.2024

**Temat:** Język opisu sceny SVG

**Wariant**: 1

Radosław Skrzypczyński

Informatyka I stopień,

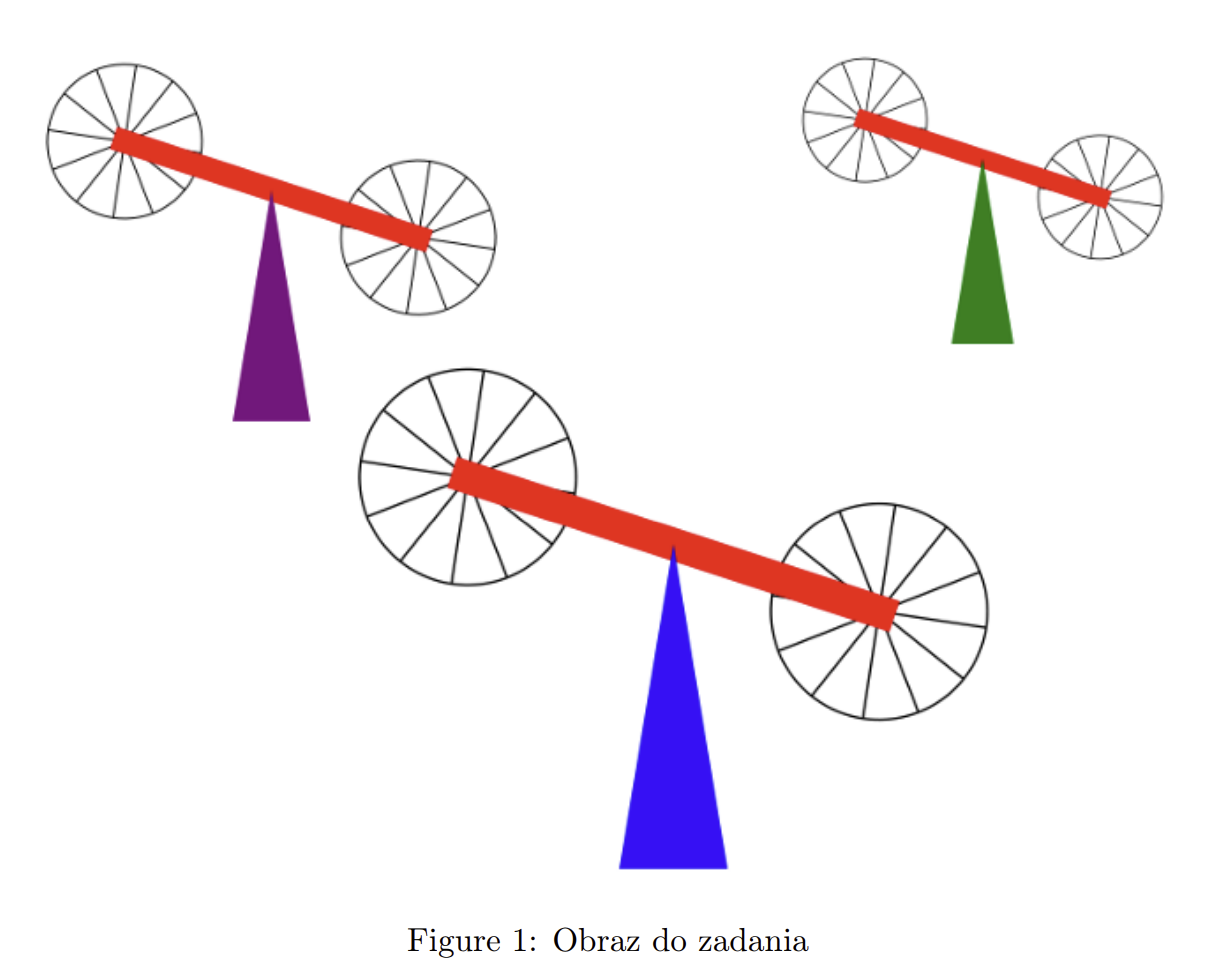
stacjonarne,

4 semestr,

Gr. 1B

1. **Polecenie:**

Opracować scenę hierarchiczną zgodnie z obrazem używając zamiast kół wielokąty obracające się (animacja!) według wariantu. Opracowanie powinno być w języku SVG.



**2. Wprowadzane dane:**

brak wprowadzanych danych

**3. Wykorzystane komendy:**

*<?xml version="1.0" encoding="utf-8"?>*

*<svg viewBox="0 0 1000 1000" xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink">*

*<defs>*

*<polygon id="pentagon" points="0,-30 28,-9 17,24 -17,24 -28,-9" style="stroke: black; fill: none;" />*

*<g id="pentagonWithLines">*

*<use xlink:href="#pentagon"/>*

*<line x1="0" y1="0" x2="0" y2="-30" style="stroke: black;" />*

*<line x1="0" y1="0" x2="28" y2="-9" style="stroke: black;" />*

*<line x1="0" y1="0" x2="17" y2="24" style="stroke: black;" />*

*<line x1="0" y1="0" x2="-17" y2="24" style="stroke: black;" />*

*<line x1="0" y1="0" x2="-28" y2="-9" style="stroke: black;" />*

*</g>*

*</defs>*

*<!-- Wiatrak 1 -->*

*<g transform="translate(120,200) scale(1.2 1.2)">*

*<polygon points="-20,130 20,130 0,0" style="fill: purple;" />*

*<g>*

*<rect x="-75" y="-5" width="150" height="10" style="fill: red;"/>*

*<g transform="translate(-75,0)">*

*<use xlink:href="#pentagonWithLines"/>*

*<animateTransform type="rotate" additive="sum" attributeName="transform" values="0;360" begin="0s" dur="3s" fill="freeze" repeatCount="indefinite" keyTimes="0; 1"/>*

*</g>*

*<g transform="translate(75,0)">*

*<use xlink:href="#pentagonWithLines"/>*

*<animateTransform type="rotate" additive="sum" attributeName="transform" values="0;360" begin="0s" dur="3s" fill="freeze" repeatCount="indefinite" keyTimes="0; 1"/>*

*</g>*

*<animateTransform type="rotate" additive="sum" attributeName="transform" values="0;360" begin="0s" dur="5s" fill="freeze" repeatCount="indefinite" keyTimes="0; 1"/>*

*</g>*

*</g>*

*<!-- Wiatrak 2 -->*

*<g transform="translate(690,180) scale(1 1)">*

*<polygon points="-20,130 20,130 0,0" style="fill: green;" />*

*<g>*

*<rect x="-75" y="-5" width="150" height="10" style="fill: red;"/>*

*<g transform="translate(-75,0)">*

*<use xlink:href="#pentagonWithLines"/>*

*<animateTransform type="rotate" additive="sum" attributeName="transform" values="0;360" begin="0s" dur="3s" fill="freeze" repeatCount="indefinite" keyTimes="0; 1"/>*

*</g>*

*<g transform="translate(75,0)">*

*<use xlink:href="#pentagonWithLines"/>*

*<animateTransform type="rotate" additive="sum" attributeName="transform" values="0;360" begin="0s" dur="3s" fill="freeze" repeatCount="indefinite" keyTimes="0; 1"/>*

*</g>*

*<animateTransform type="rotate" additive="sum" attributeName="transform" values="0;360" begin="0s" dur="5s" fill="freeze" repeatCount="indefinite" keyTimes="0; 1"/>*

*</g>*

*</g>*

*<!-- Wiatrak 3 -->*

*<g transform="translate(400,300) scale(1.6 1.6)">*

*<polygon points="-20,130 20,130 0,0" style="fill: blue;" />*

*<g>*

*<rect x="-75" y="-5" width="150" height="10" style="fill: red;"/>*

*<g transform="translate(-75,0)">*

*<use xlink:href="#pentagonWithLines"/>*

*<animateTransform type="rotate" additive="sum" attributeName="transform" values="0;360" begin="0s" dur="3s" fill="freeze" repeatCount="indefinite" keyTimes="0; 1"/>*

*</g>*

*<g transform="translate(75,0)">*

*<use xlink:href="#pentagonWithLines"/>*

*<animateTransform type="rotate" additive="sum" attributeName="transform" values="0;360" begin="0s" dur="3s" fill="freeze" repeatCount="indefinite" keyTimes="0; 1"/>*

*</g>*

*<animateTransform type="rotate" additive="sum" attributeName="transform" values="0;360" begin="0s" dur="5s" fill="freeze" repeatCount="indefinite" keyTimes="0; 1"/>*

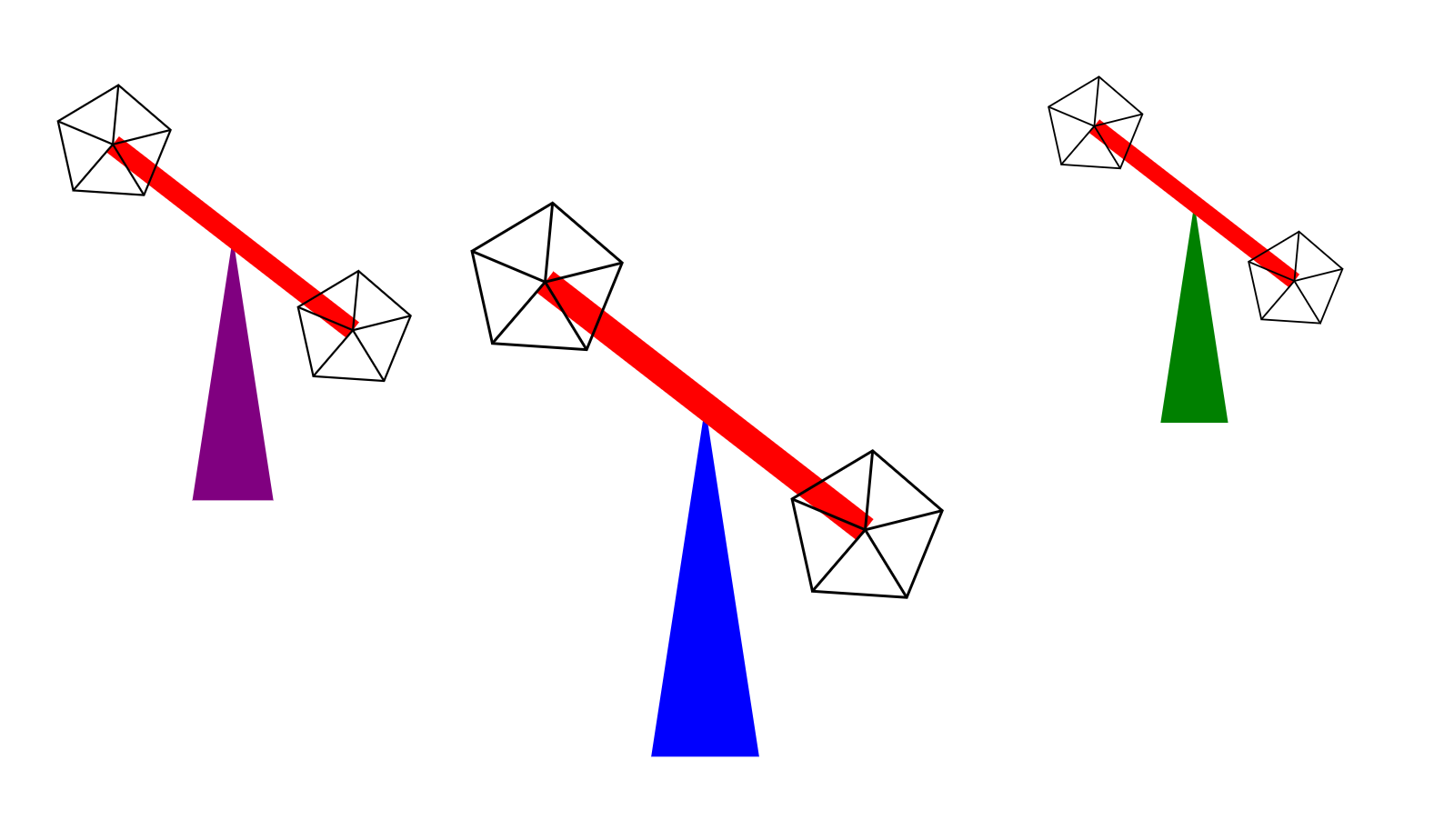
*</g>*

*</g>*

*</svg>*

Link do zdalnego repozytorium: https://github.com/Slayzerus/UBB\_GrafikaKomputerowa/tree/main/Lab%205

**4. Wynik działania:**



**5. Wnioski:**

Celem zadania było zastosowanie języka SVG do opracowania grafiki 2D, co pozwoliło na lepsze zrozumienie jego możliwości i specyfiki. SVG, jako język opisu sceny, różni się od języków programowania proceduralnych. Deklaratywna natura SVG umożliwia definiowanie zawartości sceny poprzez opisanie jej elementów, a nie przez procedury generujące te elementy.

Podczas pracy nad zadaniem, dowiedzieliśmy się, że SVG, jako język grafiki wektorowej, pozwala na precyzyjne określenie kształtów, ich atrybutów takich jak kolor i szerokość linii oraz transformacji geometrycznych. Jest to język oparty na XML, co wymaga ścisłego przestrzegania jego składni.

Struktura dokumentu SVG obejmuje elementy takie jak prostokąty, linie, koła, elipsy, tekst oraz bardziej złożone kształty jak wielokąty i ścieżki. Elementy te mogą być stylizowane za pomocą różnych atrybutów, takich jak fill, stroke, stroke-width i inne, co pozwala na uzyskanie szerokiej gamy efektów wizualnych. Transformacje takie jak rotate, scale i translate umożliwiają manipulowanie kształtami w przestrzeni.

Modelowanie hierarchiczne w SVG jest realizowane za pomocą elementu <g>, który grupuje obiekty, umożliwiając stosowanie do nich wspólnych stylów i transformacji. To podejście pozwala na traktowanie grupy elementów jako jednej jednostki, co ułatwia zarządzanie złożonymi scenami.

Podczas realizacji zadania stworzono scenę składającą się z trzech wiatraków, gdzie każdy wiatrak składa się z obracającej się czerwonej belki z pięciokątami na końcach oraz trójkąta jako podstawy. Każdy z tych elementów został umieszczony w odpowiednim miejscu na scenie, a animacje obrotu zostały dodane do czerwonych belek i pięciokątów, co umożliwiło uzyskanie dynamicznej sceny.

Zadanie pokazało, że SVG jest potężnym narzędziem do tworzenia grafiki 2D, umożliwiającym precyzyjne definiowanie kształtów, stylów i animacji. Zrozumienie i umiejętność korzystania z języka SVG jest kluczowe dla tworzenia złożonych i dynamicznych grafik wektorowych.