**SPRAWOZDANIE**

Zajęcia: Grafika komputerowa

Prowadzący: prof. dr hab. Vasyl Martsenyuk,

**Laboratorium 4**

14.04.2024

**Temat: Język opisu sceny SVG**

**Wariant 11**

Krzysztof Kłoda

Informatyka I stopień,

niestacjonarne,

4 semestr,

Gr.2/3

Polecenie :

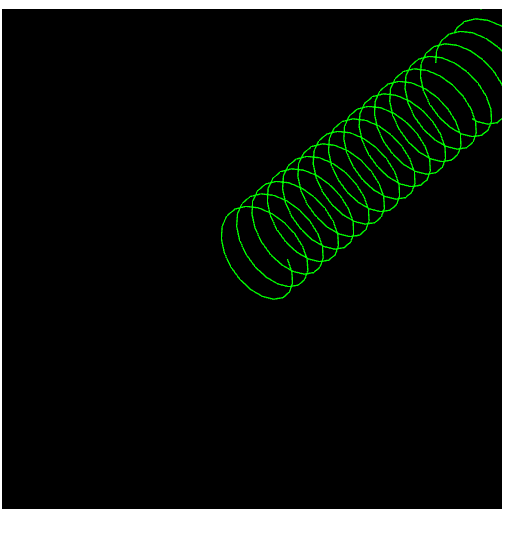
Stworzyć dwa obiekty przy użyciu OpenGL (w języku JavaScript). Po uruchomieniu zakończonego programu naciśnięcie jednego z klawiszy numerycznych 1 lub 2 spowoduje wybranie wyświetlanego obiektu. Program ustawia wartość zmiennej globalnej, objectNumber, aby powiedzieć, który obiekt ma zostać narysowany. Użytkownik może obracać obiekt za pomocą klawiszy strzałek, PageUp, PageDown i Home. Podprogram display() jest wywoływany, aby narysować obiekt.

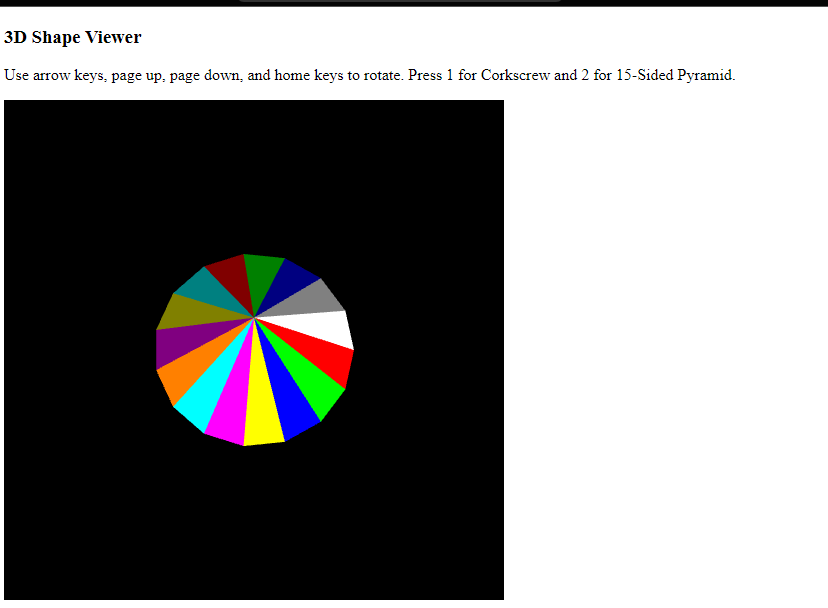
Obiekt 1. Korkociąg wokół osi {x | y | z} zawierający N obrotów. Punkty są stopniowo powiększane. Ustalić aktualny kolor rysujący na {zielony | niebieski | brązowy | … }.

Obiekt 2. Pyramida, wykorzystując dwa wachlarze trójkątów oraz modelowanie hierarchiczne (najpierw tworzymy podprogramę rysowania jednego trójkonta; dalej wykorzystując przekształcenia geometryczne tworzymy pyramidę). Podstawą pyramidy jest wielokąt o N wierzchówkach.

<https://github.com/AEZAKMI7777/GK_LAB_5>

Wynik działania :





Wnioski :

Tworzenie programów graficznych przy użyciu OpenGL w języku JavaScript pozwala na praktyczne zastosowanie wiedzy z zakresu grafiki komputerowej i interakcji użytkownika. Wykorzystanie biblioteki takiej jak glsim.js umożliwia renderowanie obiektów 3D oraz zarządzanie ich wyglądem i transformacjami. OpenGL jest popularnym API do tworzenia grafiki 3D, które umożliwia renderowanie trójwymiarowych scen.

Podczas tworzenia projektu można zdobyć wiele cennych doświadczeń. Hierarchiczne modelowanie, czyli tworzenie złożonych obiektów 3D (np. piramida), za pomocą prostych podprogramów (jak drawTriangle) i kombinowanie ich za pomocą przekształceń geometrycznych (np. glPushMatrix i glPopMatrix), pozwala na naukę efektywnego organizowania i budowania skomplikowanych struktur.

Obsługa interakcji z użytkownikiem jest kluczowa. Implementacja obsługi zdarzeń klawiszy (takich jak strzałki, PageUp, PageDown, Home) umożliwia sterowanie rotacją obiektów oraz zmianę widoczności różnych elementów sceny. To nie tylko nauka programowania interakcji w środowisku 3D, ale także sposób na integrowanie użytkownika w aplikację.

Kolejnym aspektem jest dobór kolorów i tworzenie efektów wizualnych. Ustalanie kolorów dla obiektów oraz implementacja efektów wizualnych (np. stopniowe powiększanie punktów korkociągu) jest istotne dla estetyki i czytelności renderowanej sceny. Pozwala to na eksperymentowanie z różnymi stylami i technikami artystycznymi.

Praca z OpenGL wymaga także świadomości wydajnościowej. Optymalizacja kodu oraz zrozumienie, jak OpenGL zarządza zasobami sprzętowymi, są kluczowe dla płynnej pracy programu, szczególnie przy renderowaniu skomplikowanych obiektów i efektów.

Ogółem, praca z OpenGL i JavaScriptem umożliwia rozwój szerokiego zakresu umiejętności programistycznych. Obejmuje to nie tylko grafikę komputerową, ale także zarządzanie stanem aplikacji, optymalizację wydajności oraz obsługę złożonych interakcji użytkownika.