1. Cele kursu:

* Nowoczesne techniki OpenGL,
* Utworzenie okna oraz obsługa operacji wejściowych (klawiatura),
* **Vertex**, **fragment** oraz **Geometry** **Shader**,





* Rysowanie **obiektów** **3D**,
* Używanie biblioteki **GLM** (OpenGL Maths),
* Przenoszenie (**Translate**), Obracanie (**Rotate**) oraz Skalowanie (**Scale**) modeli,
* Używanie interpolacji (**interpolation**) ~ używane do tekstur oraz światła.
* Używanie indeksowego rysowania (**indexed draws**) ~ pozwala używać wierzchołki, które już zostały wspomniane.
* Używanie różnych rodzajów projekcii/ rzutowania (**projection**) ~ ortograficzna dla 2D lub z perspektywą dla 3D.
* Kontrola **kamery** oraz poruszanie nią,
* **Mapowanie tekstur** ~ nakładanie tekstur na obiekty.
* **Phong** Model Oświetlania ~ najbardziej popularny.
* Kierunkowe (**Directional jak słońce**), Punktowe **(Point jak kula)** oraz Miejscowe **(Spot jak pochodnia)** oświetlenie.
* Importowanie wcześniej zrobionych modeli 3D.
* **Mapowanie cieni** (też z różnych źródeł światła).
* Implementacja SkyBox (iluzja dużego świata).

2. Wprowadzenie do GLEW, GLFW oraz SDL:

**GLEW:**

Co to jest GLEW? **(ROZSZERZENIA ORAZ STERUJE KARTA)**

* OpenGL Extension Wrangler **~** Obsługiwacz rozszerzeń OpenGL.
* Interfejs dla OpenGL wersji ponad 1.1
* Ładuje rozszerzenia OpenGL,
* Niektóre **rozszerzenia** są zależne od platformy, GLEW **może sprawdzić jeżeli one istnieją na tej platformie**.
* **Alternatywy:** GL3W, glLoadGen, glad, glsdk, glbinding, libepoxy, Glee,

Używanie GLEW?

* **#include <GL/glew.h>**
* Po zainicjowaniu kontekstu (GLFW) należy:
  + **glewExperimental = GL\_TRUE; (pozwala używać bardziej zawansowane operacje przy pomocy GLEW).**
  + **glewInit(); (inicjalizacja GLEW)**
* Powinno zwrócić **GLEW\_OK**. Jeżeli się nie uda to zwróci error.
* Można odczytać error używając **glewGetErrorString (result);**
* Może sprawdzić czy rozszerzenia istnieją (niektóre rozszerzenia są zależne od platformy):
  + **if(!GLEW\_EXT\_framebuffer\_object){}**
* **wglew.h** tylko dla Windows tylko z funkcjami.

**GLFW:**

Co to jest GLFW? (**TWORZY KONTEKST ORAZ INPUT) ~ KREATOR KONTEKSTU**

GLFW oraz SDL służą do tworzenia okien oraz kontekstu. **Kontekst** jest **zasadniczą maszyną stanu**, która przechowuje wszystkie dane związane z wyświetlaniem aplikacji. Gdy aplikacja jest zamykana, kontekst OpenGL jest niszczony.

* OpenGL FrameWork ~ budowa/ struktura/ ramka.
* **Obsługuje utworzenie okna (kontekstu) oraz jego kontrole (położenie, rozmiar),**
* Obsługę operacji wejściowych z klawiatury, myszy, joysticka oraz kontrolera.
* Obsługuję obsługę wielu monitorów,
* Używa OpenGL kontekst dla okien, czyli inaczej **tworzy okna** a GLEW je **wypełnia zawartością**.

**Alternatywą GLFW**, który służy do tworzenia kontekstu oraz okna **jest SDL**:

**SDL:**

* Simple DirectMedia Layer ~ prosta warstwa bezpośrednich mediów.
* Potrafi zrobić prawie wszystko co GLFW **i więcej !!!!**

**Np.: Potrafi obsługiwać:**

* + Audio,
  + Wątkowość,
  + System plików,
  + itp.,
* **Inaczej mówiąc SDL umożliwia więcej rzeczy do robienia niż tylko tworzenie kontekstu, okna i obsługę operacji wejściowych (GLFW) ale również potrafi obsługiwać audio, wątkowość oraz system plików.**
* Używane w: FTL, Amnesia, Starbound oraz Dying Light,
* Używane w edytorach poziomów dla Source Engine oraz Cryengine.

*Alternatywy dla GLFW oraz SDL:*

* **SFML** (Simple and Fast Multimedia Library): Prawie jak SDL ale zawiera więcej możliwości. Niestety kontekst OpenGL jest bardzo słaby, ponieważ bazuje na grafice 2D.
* **GLUT** (OpenGL Utility Toolkit): Należy go unikać!
* **Win32 API:** GLFW, SDL, SFML, GLUT używają tego w tle. Tylko dla osób, które wiedzą co robią. Najniższy poziom do tworzenie kontekstu/ okien. Inne kreatory kontekstu używają tego w tle.

**Podsumowanie:**

* **GLEW** (OpenGL Extension Wrangler) ~ zapewnia nam interfejs/ ***połączenie z nowoczesną wersją*** OpenGL oraz ***obsługuję rozszerzenia*** zależne platformowo (bezpiecznie).
* **GLFW** pozwala nam ***utworzyć okna*** oraz OpenGL***kontekst*** również pozwala ***obsługę operacji wejściowych*** od użytkownika (klawiatura, myszka, gamepad).
* **SDL** umożliwia wiele więcej rzeczy niż GFLW (np.: obsługę audio).

**Czyli:  
GLFW** służy do **tworzenia okna oraz kontekstu** (maszyny stanu, która przechowuje wszystkie danego związane z wyświetlaniem aplikacji).

Natomiast **GLEW** służy do **korzystania z nowoczesnej wersji** OpenGL oraz do **ładowania i korzystania z dostępnych rozszerzeń**. Dzięki niemu możemy w sposób nowoczesny korzystać z maszyn stanu. Umożliwia korzystanie z OpenGL.

GLEW umożliwia nam rysowanie kontekstu wewnątrz okna ale za to GLFW umożliwia załączenie tego kontekstu.

**Etapy załączania GLFW:**

1. Załączamy plik nagłówkowy.
2. Inicjalizujemy GLFW.
3. Ustawiamy parametry okna.