



Gry Komputerowe Tworzenie geometrii

Michał Chwesiuk

Zachodniopomorski Uniwersytet Technologiczny w Szczecinie
Wydział Informatyki

11 Kwiecień 2018

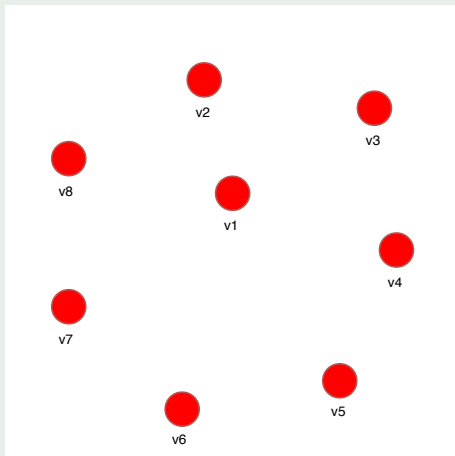


- Prymitywy graficzne są to elementarne wzorce, za pomocą których łączone są wierzchołki. Za ich pomocą można wymodelować dowolny obiekt.
- Rodzaje prymitywów w OpenGL :
 - GL_POINTS
 - GL_LINES
 - GL_LINE_STRIP
 - GL_LINE_LOOP
 - GL_TRIANGLES
 - GL_TRIANGLE_STRIP
 - GL_TRIANGLE_FAN
 - GL_QUADS
 - GL_QUAD_STRIP
 - GL_POLYGON



Rodzaje prymitywów

GL_POINTS



Prymitywy
graficzne

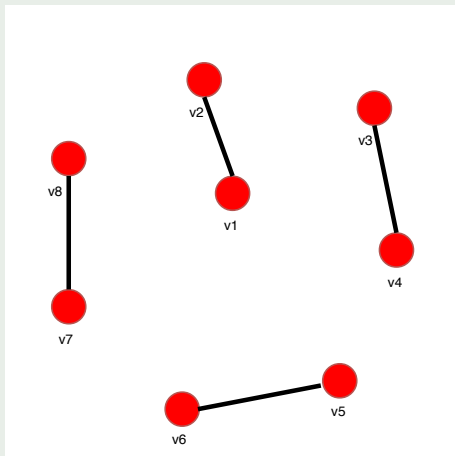
Face Culling

Zadanie



Rodzaje prymitywów

GL_LINES



Prymitywy
graficzne

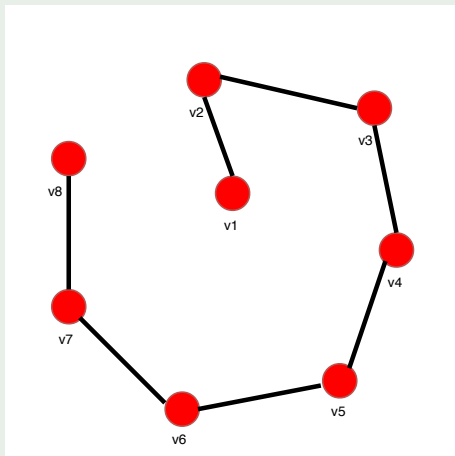
Face Culling

Zadanie



Rodzaje prymitywów

GL_LINE_STRIP



Prymitywy
graficzne

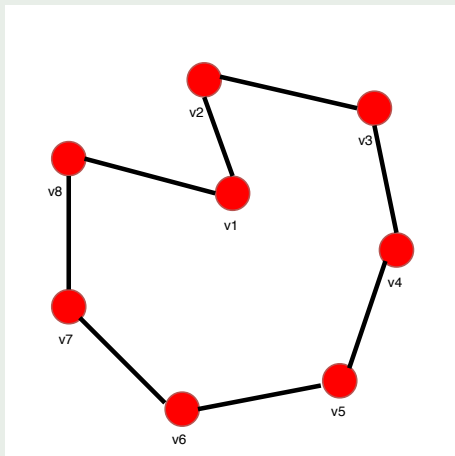
Face Culling

Zadanie



Rodzaje prymitywów

GL_LINE_LOOP



Prymitywy
graficzne

Face Culling

Zadanie



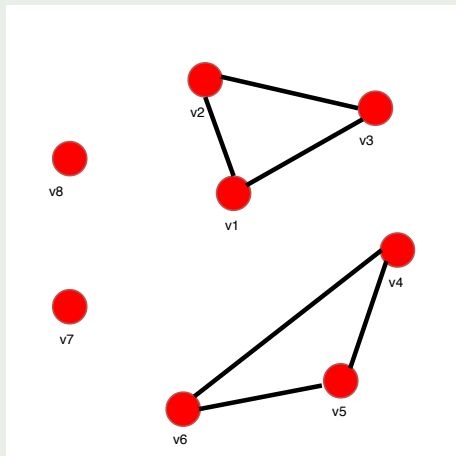
Rodzaje prymitywów

GL_TRIANGLES

Prymitywy
graficzne

Face Culling

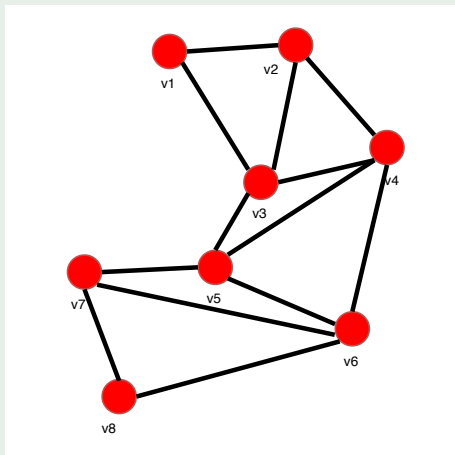
Zadanie





Rodzaje prymitywów

GL_TRIANGLE_STRIP



Prymitywy
graficzne

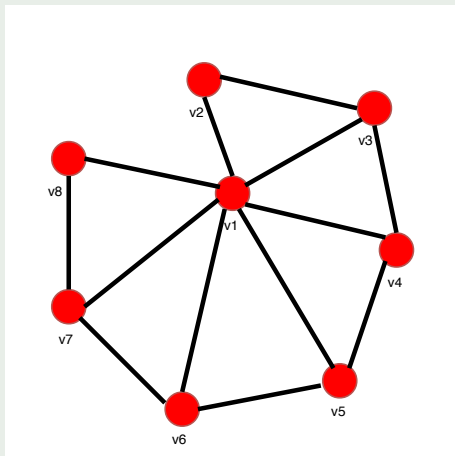
Face Culling

Zadanie



Rodzaje prymitywów

GL_TRIANGLE_FAN



Prymitywy
graficzne

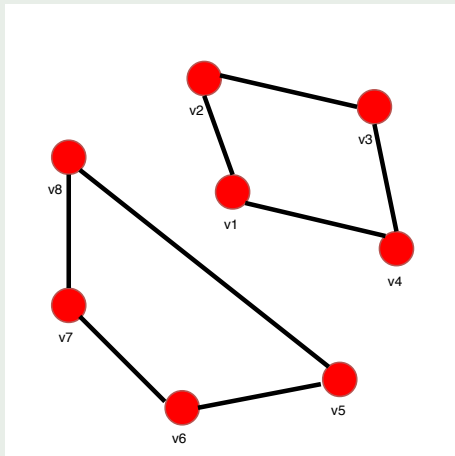
Face Culling

Zadanie



Rodzaje prymitywów

GL_QUADS



Prymitywy
graficzne

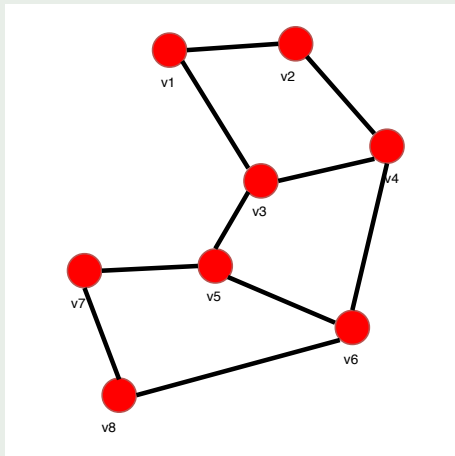
Face Culling

Zadanie



Rodzaje prymitywów

GL_QUAD_STRIP



Prymitywy
graficzne

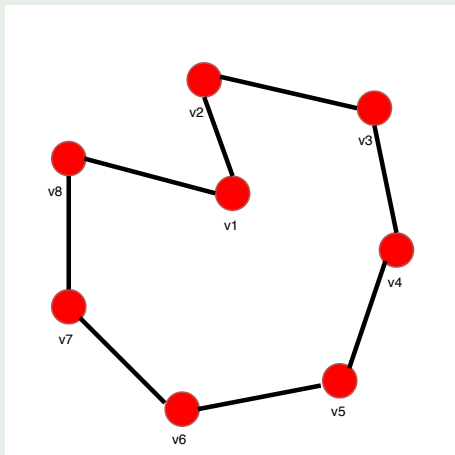
Face Culling

Zadanie



Rodzaje prymitywów

GL_POLYGON



Prymitywy
graficzne

Face Culling

Zadanie



- Geometrie w OpenGL implementuje się za pomocą poniższych funkcji :
 - *glBegin(mode)* - rozpoczęcie rysowania geometrii za pomocą prymitywu *mode*.
 - Rodzina funkcji *glVertex()* - przekazanie wierzchołka o współrzędnych przekazanych w parametach do OpenGL w celu wykorzystania go do narysowania prymitywu.
 - *glEnd()* - zakończenie rysowania aktualnego prymitywu.



Face culling

Prymitywy
graficzne

Face Culling

Zadanie

- Wypełnione prymitywy posiadają dwie ściany, z których w większości przypadków, chcemy renderować jedną.
- W celu przyśpieszenia procesu rendering, odcia się jedną ścianę (tzw. tylną). Odbywa się to za pomocą **face culling'u**.
- Face culling polega na sprawdzeniu podczas renderingu kolejności przekazania wierzchołków pod kątem tego czy (po zastosowanych transformacjach) wysłane zostały w kolejności zgodnej z ruchem wskazówek zegara czy też nie.
- Na podstawie kolejności wierzchołków i ustawionych przez programistę trybów, OpenGL kontynuuje rendering prymitywu lub nie.



Face culling - użycie

Prymitywy
graficzne

Face Culling

Zadanie

Face culling włącza się za pomocą poniższych funkcji :

- *glEnable(GL_CULL_FACE)* - włączenie face culling'u, wyłączenie za pomocą funkcji *glDisable()*.
- *glFrontFace(mode)* - ustawienie, która kolejność będzie interpretowana jako ta definiująca przednie ściany. Parametr *mode* może być ustawiony na :
 - *GL_CW* - kolejność ruch wskazówek zegara
 - *GL_CCW* - kolejność przeciwna do ruchu wskazówek zegara
- *glCullFace(mode)* - ustawienie, które ściany będą odcinane. Parametr *mode* przyjmuje wartości : *mode*przednie(*mode=GL_FRONT*) lub tylne (*mode=GL_BACK*).
 - *GL_FRONT* - odcinane będą ściany przednie
 - *GL_BACK* - odcinane będą ściany tylne



Zadanie

Prymitywy
graficzne

Face Culling

Zadanie

Wersja Podstawowa

Zadaniem na dzisiejszych zajęciach jest zaimplementowanie własnej niestandardowej bryły geometrycznych:

- Bryła powinna zawierać krawędzie "ostre" i "gładkie"
- Poprawny rendering przy włączony face culling'u
- Wstawienie kilku instancji zaimplementowanej bryły

Wersja Rozszerzona

Do poruszania kamerą należy dodać :

- Nie zależny ruch części bryły
- Wstawienie wielu instancji bryły w efektywny sposób, czyli bez każdorazowego przekazywania współrzędnych wierzchołków (np. przy użyciu *Display List*, *Vertex Array*, *VBO*)